

Ein effizienter gekoppelter FEM-SBFEM Ansatz zur Analyse von Boden-Bauwerk-Interaktionen im Zeitbereich

Marco Matthias Schauer

**Institut für Statik
Technische Universität Braunschweig**

Bericht Nr. 2015-117



Ein effizienter gekoppelter FEM-SBFEM Ansatz zur Analyse von Boden-Bauwerk-Interaktionen im Zeitbereich

Von der
Fakultät Architektur, Bauingenieurwesen und Umweltwissenschaften
der Technischen Universität Carolo-Wilhelmina
zu Braunschweig

zur Erlangung des Grades eines
Doktoringenieurs (Dr.-Ing.)
genehmigte

Dissertation

von
Marco Matthias Schauer
geboren am 13. Juli 1981
aus Eckernförde

Eingereicht am: 14. November 2014

Disputation am: 17. Dezember 2014

Berichterstatter/in: Prof. Dr.-Ing. S. Langer
Prof. Dr.-Ing. D. Dinkler

2015

ISBN 978-3-926031-18-1

Herausgeber: Prof. Dr.-Ing. Dieter Dinkler

©Institut für Statik, Technische Universität Braunschweig, 2015

Kurzfassung

Bauwerke sind auf den umgebenden Baugrund gegründet, sodass Baugrund und Bauwerk miteinander interagieren können. Dabei können in den Baugrund eingeleitete Erschütterungen in die angrenzenden Gebäude übertragen werden. Zusätzlich interagieren benachbarte Bauwerke, wenn diese über den Baugrund miteinander verbunden sind. Heute ist die erdbebensichere Auslegung von Strukturen in seismisch aktiven Regionen oder auch die Komfortsteigerung von Gebäuden, indem diese von umgebenen Emissionen wie z. B. eingebrachten Vibrationen durch Verkehr oder Maschinenfundamente entkoppelt werden, von großem Interesse. Bei der Bearbeitung dieser sehr unterschiedlichen Fragestellungen wird immer häufiger auf numerische Simulationen zurückgegriffen, um die Boden-Bauwerk-Interaktion zu untersuchen.

In dieser Arbeit wird gezeigt, dass die Simulation der Boden-Bauwerk-Interaktion unter Berücksichtigung randloser Gebiete, die die Sommerfeld'sche Abstrahlbedingung exakt erfüllen, nicht nur für akademische Beispiele, sondern auch für großskalige realitätsnahe Modelle, wie sie in der Praxis benötigt werden, möglich ist. Dafür wird mit zwei numerischen Methoden ein effizientes gekoppeltes Verfahren zur Simulation der Boden-Bauwerk-Interaktion im Zeitbereich vorgeschlagen. Die numerische Umsetzung des gekoppelten Verfahrens beruht auf einer Kombination von Finite Elemente Methode und *Scaled Boundary Finite Element Method*. Die Finite Elemente Methode bildet dabei das Nahfeld ab, in dem die zu untersuchende Struktur samt anstehenden Baugrund enthalten ist. Der angekoppelte unendliche Halbraum wird als Fernfeld mit der *Scaled Boundary Finite Element Method* diskretisiert.

Weil die Simulation der Boden-Bauwerk-Interaktion im Zeitbereich mit einem großen Rechenaufwand und Speicherbedarf einhergeht, werden unterschiedliche Methoden eingesetzt um eine numerische Simulation in angemessener Zeit durchführen zu können. Es wird gezeigt, dass bei entsprechender Diskretisierung des Fernfeldes realitätsnahe Fragestellungen der Boden-Bauwerk-Interaktion untersucht werden können. Verschiedene Ansätze werden für die Reduktion des Berechnungsaufwands verfolgt und miteinander kombiniert. Zusätzlich erfolgt eine Parallelisierung der Berechnungsalgorithmen.

Nach Einführung in die theoretischen Grundlagen der Modellbildung und einer Diskussion der numerischen Verfahren wird auf die Validierung des hier gewählten Kopplungsansatzes eingegangen. Dies erfolgt anhand ausgewählter Beispiele, für die analytische bzw. semi-analytische Lösungen bekannt sind. Anschließend wird die Effizienz der implementierten Software mittels einer Analyse der Leistungsfähigkeit der parallelen Algorithmen getestet. Abschließend werden mögliche Anwendungen aus dem Bereich der Geotechnik vorgestellt und die Anwendbarkeit des hier entwickelten Verfahrens an Modellbeispielen gezeigt.

Abstract

Buildings are directly in contact with surrounding, such that the soil and the structure interact with one other. Consequently, soil induced vibrations are transmitted to the structures. Additionally nearby structures interact with one another as they are connected by the soil. Nowadays numerical simulation of soil-structure-interaction is of great interest, and is applied to a wide range of different problems. These include the analysis and design of reliable earthquake-resistant structures in seismic active areas, and also design to the increase the comfort of buildings by decoupling them from surrounding emissions such as vibrations induced by traffic of machine foundations.

The present work shows the simulation of soil-structure-interaction which takes unbounded domains into account. This work fulfils the Sommerfeld radiation condition exactly, and shows that it is not only applicable for academic examples, but for large scale real life problems as well. Two numerical methods are coupled to create an efficient coupled method which can be used to simulate soil-structure-interaction in the time domain. The numerical implementation of this coupled approach is based on a combination of the finite element method and the scaled boundary finite element method. The finite element method is used to discretise the near field, containing structures and its surrounding soil. The coupled infinite half-space the so called far field is realised by the scaled boundary finite element method.

The simulation of soil-structure-interaction in the time domain is computationally very time and memory intensive. Different methods are used to perform numerical simulations in the appropriate time. It is shown that using a suitable optimisation of the far field, realistic problem analysis of the soil-structure-interaction can be realised, therefore various optimisation approaches are used and combined with each other. Additionally a contemporary parallel implementation of the algorithms is done.

After introducing the theoretical background and discussing the chosen numerical approach, a validation of the used coupling scheme is done. This validation is carried out by comparison of numerical and analytic solutions for defined test cases. Subsequently the numerical performance of the implemented software is tested in terms of speed-up and efficiency. Finally, different geotechnical applications are illustrated and the applicability of the coupled method is shown and discussed using examples.

Inhaltsverzeichnis

1. Einleitung	1
1.1. Stand der Technik	3
1.2. Ziel der Arbeit	7
2. Theoretische Grundlagen	9
2.1. Berechnung und Darstellung von Spannungen	10
2.2. Stoffmodelle	11
2.3. Wellenausbreitung	13
2.4. Sommerfeld'sche Abstrahlbedingung	15
3. Numerische Modellbildung	17
3.1. Finite Elemente Methode	17
3.2. Zeitschrittintegrationsverfahren	20
3.2.1. Newmark-Verfahren	20
3.2.2. Hilber-Hughes-Taylor- α -Verfahren	21
3.2.3. Generalized- α -Verfahren	21
3.2.4. Allgemeiner Lösungsansatz	22
3.2.5. Parameterwahl	24
3.2.6. Zeitschrittlänge	25
3.2.7. Energiebilanz	25
3.3. <i>Scaled Boundary Finite Element Method</i>	26
3.3.1. Frequenzbereich	27
3.3.2. Zeitbereich	28
3.3.2.1. Erster Zeitschritt	29
3.3.2.2. n -ter Zeitschritt	30
3.4. Kopplung von FEM und SBFEM	30
4. Wissenschaftliches Rechnen	33
4.1. Paralleles und verteiltes Rechnen	34
4.2. Softwareimplementierung	36
4.2.1. <i>Scaled Boundary SCABO</i>	37
4.2.2. <i>elementary Parallel Solver</i> ELPASO	40
4.2.3. FEM-SBFEM Schnittstelle	41
5. Validierung des gekoppelten Ansatzes	45
5.1. Balken	45
5.1.1. Numerische Lösung	47
5.1.2. Analytische Lösung	51
5.1.3. Numerische Lösung des gekoppelten Verfahrens	53

5.2. Setzung	57
5.2.1. Semi-analytischer Lösungsansatz	57
5.2.2. Numerische Lösung	58
6. Effizienzsteigerung des gekoppelten Ansatzes	63
6.1. Extrapolation der Einflussmatrizen	63
6.2. Geometrische Entkopplung	68
6.3. Zerlegung des Fernfeldes in Substrukturen	70
6.4. Fazit	74
7. Parallelisierung der diskretisierten Modellgleichungen	75
7.1. Amdahls und Gustafsons Gesetze	75
7.2. SCABO: Fernfeldberechnung	79
7.2.1. Starke Skalierung	79
7.2.2. Schwache Skalierung	80
7.2.3. Vergleich der Lösungsansätze	82
7.3. ELPASO: Gekoppelter-Nahfeld-Fernfeld-Ansatz	83
7.4. Abschätzung des sequentiellen Programmanteils	86
8. Anwendungen in der Geotechnik	89
8.1. Statische und dynamische Belastung eines Fundaments	91
8.1.1. Verschiebungen und Spannungen unter Fundamenten	93
8.1.2. Dynamische Belastung eines Fundaments	96
8.2. Dynamische Belastung von Pfählen	98
8.2.1. Pfahl-Integritätstests an Modellpfählen	98
8.2.1.1. Numerische Simulation	100
8.2.1.2. Ergebnisse	102
8.2.2. In situ Rammtest	105
8.3. Dynamische Lasten aufgrund von Verkehr	108
8.3.1. Modellbildung	108
8.3.2. Ergebnisse	110
9. Zusammenfassung und Ausblick	115
A. Nomenklatur	i
Literaturverzeichnis	iii

1. Einleitung

In der urbanen Gesellschaft nimmt der Komfort und die Standsicherheit von Gebäuden eine zentrale Rolle ein. Aufgrund der sehr hohen Wohndichte in den Großstädten und Ballungszentren und dem stetig zunehmenden Verkehr innerhalb dieser Räume steigt gleichzeitig die durch den Verkehr verursachte Belastung der Bewohner. Vorbeifahrende Fahrzeuge wie beispielsweise Züge, Straßen- oder U-Bahnen, LKW und Busse verursachen teils unangenehme Vibrationen innerhalb der angrenzenden Gebäude.

Messungen der Erdbebenstation Bensberg der Universität zu Köln belegen, dass die neue U5 den Kölner Dom zum Schwingen anregt [51, 54]. In einer Pressemitteilung vom 14. Januar vergangenen Jahres informierte die Kölner-Verkehrs-Betriebe AG über das aktuelle Vorgehen und mögliche Ursachen [91]. Auch der Mensch kann aufgrund rhythmischer Bewegungen Gebäude in Schwingungen versetzen und sogar zum Einsturz bringen. So stürzte z. B. 1831 die *Broughton Suspension Bridge* zwischen *Broughton* und *Pendleton* in Großbritannien ein, als 60 Soldaten diese im Gleichschritt marschierend überquerten [160]. Neben dynamischen Anregungen, die durch den Verkehr verursacht werden, können auch Einwirkungen, die durch industriell genutzte Maschinen über das Maschinenfundament in den Baugrund eingetragen werden, ähnliche Effekte verursachen.

Das Zusammenwirken von Boden und Bauwerk wird als Boden-Bauwerk-Interaktion bezeichnet: Bauwerke stehen in direktem Kontakt mit dem Baugrund, sodass Baugrund und Bauwerk zwangsläufig miteinander interagieren. Dies hat zur Folge, dass in den Baugrund eingeleitete Erschütterungen in angrenzende Gebäude übertragen werden können. Des Weiteren kann es zu Interaktionen zwischen benachbarten Bauwerken kommen, wenn diese über den Baugrund miteinander verbunden sind. Werden Erschütterungen innerhalb eines Gebäudes verursacht, können diese im Einzelfall auch in den Nachbargebäuden vom Menschen wahrgenommen werden. Anregungen mit einer Frequenz von 1 - 80 Hz führen zu vom Menschen spürbaren Erschütterungen, wobei die Frequenz und die Größe der Amplitude eine entscheidende Rolle spielen [90]. Im Bereich von 16 Hz bis 20 kHz kann es zu hörbarem Luftschall kommen [47]. Derartige Vibrationen werden in der Regel messtechnisch erfasst und ausgewertet.

Eine Prognose und Beurteilung der eingetragenen Erschütterungen ist notwendig, um die Gebrauchstauglichkeit der angrenzenden Gebäude sicherstellen zu können. Es gilt Infrastrukturbawerke und laufende Maschinen so abzustimmen, dass die Stabilität des Bauwerks sichergestellt ist. Darüber hinaus gilt es störende Vibrationen in der Nachbarbebauung zu vermeiden und damit den Komfort der Nutzer zu erhöhen. Aus dem daraus resultierenden Sachverhalt müssen Emission, Transmission und Immission von Energie betrachtet werden. Wird der direkte Luftschall vernachlässigt, steht am Anfang der dreigliedrigen Kette der Emittent, durch den die Erschütterungen verursacht werden. Darauf folgt die Transmission der eingeleiteten Energie durch den Baugrund bis hin zum Immitenten. In dem so angeregten Bauwerk werden die Erschütterungen für den Menschen spürbar und bei entsprechender Anregung einzelner Bauteile gegebenenfalls sogar hörbar.

Für die Simulation der Boden-Bauwerk-Interaktion zur dynamischen Untersuchung von Strukturen ist ein vollständiges Boden-Bauwerk-Modell erforderlich, damit die auftretenden Effekte beschrieben werden können [4, 115, 122]. Einfache Rechenmodelle, die idealisierte Masse-Feder-Dämpfer-Modelle abbilden, reichen in der Regel nicht aus. Ziel der Simulation ist eine detaillierte und physikalisch korrekte Abbildung des Bauwerkverhaltens inklusive des Untergrunds. Bei der Modellierung des anstehenden Bodens ist zu beachten, dass es sich um einen unendlichen Halbraum handelt, in den die Struktur eingebettet ist. In den numerischen Standardverfahren ist die Abbildung unendlicher Gebiete zumeist nicht möglich, weshalb künstliche Ränder eingeführt werden müssen. Die künstlichen Ränder, die durch die gewählte Diskretisierung auftreten, sind so zu definieren, dass sie keinen oder zumindest einen vernachlässigbar kleinen Einfluss auf die Lösung haben. Dies ist besonders wichtig für Simulationen im Zeitbereich, damit auf den Gebietsrand treffende Wellen nicht reflektiert werden und so die Lösung verfälschen. Randbedingungen, die die Sommerfeld'sche Abstrahlbedingung erfüllen (vgl. 2.4), verhindern die Reflexion eintreffender Wellen am Gebietsrand vollständig.

Aktuell existieren eine Reihe von Verfahren, die zur Simulation derartiger Problemstellungen genutzt werden können. Hierzu zählen die Finite Elemente Methode (FEM), die Randelementemethode oder auch *Boundary Element Method* (BEM), die Integraltransformationemethode (ITM) sowie die *Scaled Boundary Finite Element Method* (SBFEM). Um die jeweiligen Vorteile der einzelnen Methoden zu nutzen und Nachteile zu kompensieren, werden im Einzelfall unterschiedliche Verfahren miteinander gekoppelt.

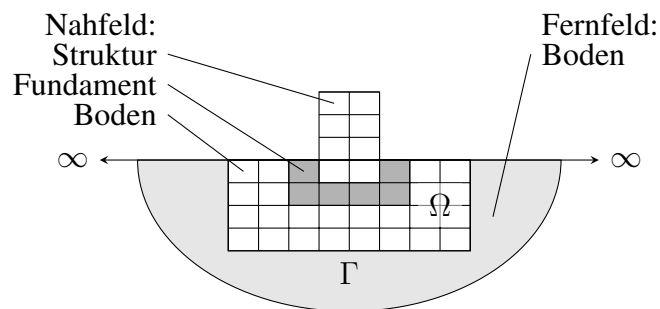


Bild 1.1.: Schematische Darstellung von Nahfeld Ω , Grenzfläche Γ und angrenzendem Fernfeld.

Aufgrund der vorliegenden Phänomenologie der Boden-Bauwerk-Interaktion ist es sinnvoll, das Gebiet in zwei Teilstrukturen zu zerlegen. Zum einen in ein Nahfeld Ω , das die Struktur sowie den anstehenden Baugrund beinhaltet. Zum anderen in ein Fernfeld, das den umgebenden unendlichen Halbraum beschreibt. Beide Teilstrukturen werden aufgrund ihrer Phänomenologie mit unterschiedlichen Modellen beschrieben und entsprechenden numerischen Verfahren diskretisiert. Bild 1.1 skizziert die Aufteilung in Nah- und Fernfeld. Die Kopplung der beiden Teilstrukturen erfolgt über den gemeinsamen Rand Γ .

1.1. Stand der Technik

Die Entwicklung numerischer Methoden zur Simulation der Boden-Bauwerk-Interaktion beginnt mit den Arbeiten von FREEDMAN und WONG in den siebziger Jahren [58, 164]. In den Folgejahren werden für unterschiedliche Fragestellungen diverse Methoden wie FEM, BEM, SBFEM oder auch die Integraltransformationsmethode (ITM) entwickelt. Hinzu kommen gekoppelte Verfahren wie FEM-BEM, FEM-SBFEM, FEM-ITM und andere Methoden, die entwickelt werden, um die Vorteile der verschiedenen Verfahren miteinander zu kombinieren und somit gleichzeitig Nachteile anderer Verfahren zu kompensieren. MÜLLER [119] und SHAH et al. [145] liefern einen differenzierten Überblick.

Bei der Kopplung wird örtlich und zeitlich lokal bzw. nicht-lokal unterschieden. Örtlich lokale Kopplungen nutzen Knoteninformationen ausschließlich lokal, das heißt, es werden Knoteninformationen lediglich an den direkten Kopplungspunkten zwischen den unterschiedlichen Modellen ausgetauscht. Nicht-lokale Kopplungen beziehen Nachbarknoten oder auch den gesamten Kopplungsrand mit ein. Ähnlich verhält es sich mit der zeitlich lokalen Kopplung, hier werden lediglich Informationen des aktuellen Zeitschritts zwischen den Modellen ausgetauscht. Bei zeitlich nicht-lokaler Kopplung werden auch Informationen bezüglich der vorangegangenen Zeitschritte berücksichtigt.

Die gewählte Kopplungsstrategie entscheidet maßgeblich über die Parameter, die zur Kopplung der unterschiedlichen Methoden genutzt werden. Hierzu zählen, abhängig von der Kopplung, Verschiebungen, Geschwindigkeiten, Beschleunigungen, Kräfte und Spannungen; gegebenenfalls sind weitere Größen erforderlich. Bezüglich der Art der Kopplung, direkt oder iterativ, unterscheiden sich die Verfahren ebenfalls. Bei der direkten Kopplung wird das Lösungsverfahren so aufgebaut, dass die Modellgleichungen gleichzeitig innerhalb eines Lösungsschritts gelöst werden. In diesem Fall wird häufig von einer monolithischen Kopplung gesprochen, wenn die Gleichungssysteme zu einem Gesamtsystem zusammengebaut werden. Bei der iterativen Kopplung hingegen wird jedes System separat gelöst. Im Anschluss daran erfolgt die Übergabe der Koppelparameter zwischen den einzelnen Modellen und wird bei der neuen Lösung berücksichtigt. Sind beide Gleichungssysteme erfüllt, ist die gesuchte Lösung gefunden. Anderenfalls erfolgt ein erneuter Austausch der Koppelparameter.

Finite Elemente Methode Die FEM ist bei Analysen strukturdynamischer Fragestellungen das am weitesten verbreitete Verfahren. Die FEM liefert eine Näherungslösung der zugrunde liegenden Differentialgleichung. Dabei wird das betrachtete Gebiet in Elemente finiter Größe aufgeteilt und an den über die Ansatzfunktionen gewählten Stützstellen, den Elementknoten, diskret gelöst. Ausführliche Informationen zur FEM und deren Umsetzung für unterschiedliche Problemstellungen sind unter anderem in [14] und [171] zu finden. Für die Modellierung unendlicher Gebiete, wie sie für die Boden-Bauwerk-Interaktion benötigt werden, ist sie jedoch ungeeignet. Dies liegt in erster Linie daran, dass nur Netze einer endlichen Ausdehnung diskretisiert werden können.

Für die Simulation der Boden-Bauwerk-Interaktion werden eine Reihe von durchlässigen, teildurchlässigen und nicht-reflektierenden Randbedingungen entwickelt, um das Verhalten des unendlichen Halbraumes mit Hilfe der FEM zu approximieren. Das einfachste Modell realisierten LYSMER und KUHMEIER bereits 1969, indem sie die geometrische Dämpfung des unendlichen Gebiets mit einem Dämpfungselement auf dem Rand berücksichtigten [112]. Weitere nicht-reflektierende Randbedingungen wurden unter anderem von ENGQUIST und MAJDA [53], LIAO und WONG [105], GROTE und KELLER [73] oder auch APPELÖ und COLONIUS [5] beschrieben. Eine umfangreiche Zusammenstellung verschiedener nicht-reflektierender Randbedingungen mit ausführlicher Diskussion liefert GIVOLI [66,67]. 1974 haben VAISH und CHOPRA die Substrukturmethode eingeführt und unter Einsatz zweidimensionaler Simulationen im Frequenzbereich gezeigt, dass es möglich ist, das Fernfeld unabhängig vom Nahfeld zu berechnen und erst im Anschluss die Steifigkeitseigenschaften des Fernfeldes mit dem Nahfeld zu koppeln [157]. Auf Grundlage dieses Ansatzes sind weitere Verfahren entwickelt worden.

Randelementmethode Als Alternative zur FEM wird häufig die REM (oder *Boundary Element Method* BEM) genannt. Mit dieser Formulierung wurden im wesentlichen zwei Dinge erreicht: Zum einen ist das Erstellen der Netze mit der BEM einfacher, da die Modellgleichungen lediglich auf dem Rand diskretisiert werden und somit das gesamte Netz um eine Dimension reduziert wird. Dies bedeutet, dass Volumen durch Flächen, Flächen durch Linien und Linien durch Punkte abgebildet werden. Zum anderen ist das zu lösende Gleichungssystem aufgrund der reduzierten Anzahl an Knoten und Freiheitsgraden deutlich kleiner. Die Matrizen sind jedoch vollbesetzt und unsymmetrisch, wodurch die Auswahl der Gleichungslöser stark eingeschränkt wird.

Die Sommerfeld'sche Abstrahlbedingung erfüllt die BEM im Gegensatz zur FEM aufgrund der Fundamentallösung exakt. Diese muss allerdings für das jeweilige Problem bekannt sein. Hinzu kommt, dass in der Regel nur homogene Gebiete betrachtet werden können. Für die Untersuchung inhomogener oder heterogener Gebiete ist ein erheblicher Mehraufwand zu betreiben. Dabei muss das inhomogene Gebiet im ersten Schritt in homogene Teilgebiete zerlegt werden. Im zweiten Schritt werden die einzelnen Teilgebiete miteinander gekoppelt. Die Grundlagen der Methode wurden unter anderem von BEER [21], BREBBIA et al. [37], DOMINGUEZ [48] und GAUL et al. [62] zusammengestellt und ausführlich erläutert. Des Weiteren ist in der genannten Literatur eine zeitliche Einordnung sowie Entwicklung der BEM zu finden. Einen guten Überblick über die unterschiedlichen Anwendungsbereiche der BEM stellen BESKOS et al. zusammen [26].

Während BESKOS und MANOLIS Tunnel und Fundamente unter statischer und dynamischer Beanspruchung mit Hilfe der BEM untersuchten [114], kam die gleiche Methode bei DOMINGUEZ zur Analyse der Boden-Bauwerk-Interaktion sowie der Interaktion von Dammbauwerk, Boden und Wasserreservoir zum Einsatz [48]. LATZ simuliert auf Grundlage der BEM die Interaktion von flüssigkeitsgefüllten Behältern mit dem umgebenden Boden [94]. KLEIN betrachtet die Abschirmung von Tragwerken mit Hilfe von Schlitten im Baugrund [87].

FEM-BEM Die Modellierung komplexer Geometrien mit der BEM ist sehr aufwändig, sodass direkte und iterative Ansätze zur Kopplung von FEM und BEM entwickelt wurden. ZIENKIEWICZ, KELLY und BETTESS gehören zu den Ersten, die sich mit der Kopplung von FEM und BEM befassen [169]. Mit einem direkt gekoppelten FEM-BEM Modell haben PRABUCKI und VON ESTORFF [126] die Antwort des unendlichen Halbraums auf dynamische Einwirkungen im zweidimensionalen und DE ARAÚJO [46] im dreidimensionalen Raum untersucht. RENAULT entwickelt auf Grundlage einer direkten FEM-BEM Kopplung ein Bewertungsverfahren zur Beurteilung der Erdbebensicherheit von Brückenbauwerken [130].

Die direkte Kopplung von FEM und BEM führt zu einer Matrix, die aus zwei mal zwei Untermatrizen aufgebaut ist. Die Hauptdiagonalen der Matrix folgen aus der FEM- sowie der BEM-Diskretisierung. Die Matrizen der Nebendiagonalen realisieren die Kopplung der Freiheitsgrade von Nah- und Fernfeld. Bei großen Systemen hat die Gesamtmatrix eine hohe Konditionszahl, die sich ungünstig auf das Lösungsverhalten auswirkt. Die Bandstruktur und Symmetrie der FE-Matrix geht aufgrund der zusätzlichen BE-Matrix sowie der Kopplungseinträge verloren.

Einen iterativen Dirichlet-Neumann Algorithmus stellen LIN et al. zur Kopplung von FEM und BEM vor [106]. Sie tauschen zwischen den Substrukturen Verschiebungen und Kräfte aus, wobei das Konvergenzkriterium bezüglich der Verschiebungen formuliert ist. Ein nicht-linearer Ansatz, der im Nahfeld nicht-lineares Materialverhalten berücksichtigt, wird von VON ESTORFF et al. vorgestellt [159]. SOARES, MANSUR und VON ESTORFF diskutieren die Effizienz ihres direkten Kopplungsansatzes, der ein separates Lösen jeder einzelnen Substruktur ermöglicht, ohne die bis dahin erforderliche Iteration zwischen den Strukturen durchzuführen [147].

Infinite Elemente Methode In Anlehnung an die FEM wird die Infinite Elemente Methode (IEM) von BETTESS und ZIENKIEWICZ entwickelt und beschrieben [28]. Es handelt sich dabei im Wesentlichen um finite Elemente, die in Richtung des unendlichen Raumes spezielle Ansatzfunktionen haben, um die auslaufenden Wellen zu dämpfen und eine Reflexion ins Nahfeld zu verhindern. Einen Überblick über diese Methode sowie Anwendungen liefert BETTES [27]. ASTLEY untersucht die IEM hinsichtlich Genauigkeit, Stabilität und Leistungsfähigkeit [6]. Mit einem gekoppelten Ansatz aus FEM und IEM untersucht MAHRAN die Boden-Bauwerk-Flüssigkeit-Interaktion aufgrund dynamischer Anregung [113]. Eine Alternative zu den Infinite Elementen stellen die *Perfectly Matched Layers* PML dar. Dabei wird das Material schichtweise angepasst, um eine Reflexion der Welle in das Nahfeld zu verhindern [12, 13, 81].

Integral Transfer Methode Ein weiteres Verfahren ist die ITM, die es ebenfalls ermöglicht, den unendlichen Halbraum abzubilden. MÜLLER und FRÜHE untersuchen mit Hilfe eines gekoppelten FEM-ITM Ansatzes die dynamische Tunnel-Halbraum-Interaktion [59, 119], indem sie das Durchfahren eines Tunnels mit einem Zug simulieren. Im Falle der Tunnel-Boden-Bauwerk-Interaktion erfolgt die Modellierung von Tunnel und Bauwerk mit der FEM. Beide Strukturen sind dabei unabhängig voneinander und lediglich über die ITM, mit der der Boden

beschrieben wird, gekoppelt. In beiden Arbeiten ist eine ausführliche Literaturrecherche bezüglich der numerischen Simulation von vorbeifahrenden Zügen zu finden.

Scaled Boundary Finite Element Method Die gemeinsam von WOLF und SONG entwickelte SBFEM [161, 162] wird ursprünglich speziell für die Simulation der Boden-Bauwerk-Interaktion und geomechanische Untersuchungen entwickelt [15, 163], später kommen weitere Anwendungsgebiete wie z. B. die Untersuchung strukturdynamischer Fragestellungen [149], die Berechnung elektromagnetischer Felder [108, 109] und die Simulation von Rissausbreitungsprozessen [41, 150, 167] hinzu. Es handelt sich um eine semi-analytische Methode, bei der ähnlich wie bei der Randelementemethode lediglich der Rand des betrachteten Gebiets zu diskretisieren ist. Wie schon die BEM erfüllt auch die SBFEM die Sommerfeld'sche Abstrahlbedingung implizit und eignet sich somit ebenfalls zur Beschreibung des unendlichen Halbraums. Ausgehend vom Skalierungszentrum verwendet die Methode einen analytischen Lösungsansatz. Entlang des Randes entspricht das Lösungsverfahren dem der Finite Elemente Methode. Details hierzu werden in Abschnitt 3.3 eingehend diskutiert. Gegenüber der BEM hat die SBFEM einige Vorteile. Zum einen wird keine Fundamentallösung benötigt und das Lösen der singulären Integrale entfällt ebenfalls. Zum anderen sind die Systemmatrizen symmetrisch, was sich positiv auf das Berechnungsverfahren auswirkt.

Eine Umsetzung der SBFEM im Zeitbereich mit Hilfe der fraktionalen Infinitesimalrechnung diskutiert TRINKS in ihrer Dissertation [156]. Dieser Ansatz wird von TRINKS und SONG in den Folgejahren kontinuierlich erweitert und weiterentwickelt [40].

Der in dieser Arbeit eingesetzte Lösungsansatz ist weder lokal in der Zeit noch im Raum, was bedeutet, dass räumliche Informationen eines Punktes, wie z. B. Verschiebungen oder Kräfte, von anderen benachbarten Punkten abhängig sind und sich somit gegenseitig beeinflussen. Das selbe gilt für die Zeit: Informationen zum aktuellen Zeitpunkt bauen auf vorangegangenen Zeitschritten auf. Die SBFEM ist somit nicht-lokal in Zeit und Raum. Aufgrund der nicht-lokalen Eigenschaften benötigt die Methode große Datenmengen zum Speichern der Zustandsgrößen jedes einzelnen Zeitschrittes. Hinzu kommen nahezu vollbesetzte Matrizen, um die räumliche Kopplung abbilden zu können.

FEM-SBFEM Aufgrund der Tatsache, dass mit der SBFEM nur relativ einfache Netze erzeugt werden können, koppeln WOLF und SONG [162] und später auch andere die FEM und die SBFEM, um komplexere Geometrien im Nahfeld abbilden zu können. BORSUTZKY untersucht mit einem gekoppelten FEM-SBFEM Ansatz die Verformung von Versorgungsleitungen aufgrund von Erdbebenlasten [32, 33]. LIN et al. untersuchen den Erdbebeneinfluss auf Staumauern [107]. Ein gekoppeltes Modell, das das nicht-lineare Materialverhalten des Nahfeldes berücksichtigt, wird von CROUCH und BENNETT [44] sowie BRANSCH und LEHMANN [34–36] vorgestellt.

Aufgrund des großen numerischen Aufwands werden verschiedene Optimierungsversuche diskutiert. Zum einen wird die Zerlegung des Fernfeldes in Substrukturen von BORSUTZKY [32] sowie SCHAUER und LEHMANN [139] diskutiert, zum anderen die Linearisierung der Einflussmatrizen, die LEHMANN et al. erfolgreich eingesetzt haben [96, 99]. Des Weiteren haben LEHMANN [97] sowie LEHMANN und BORSUTZKY [100] die Möglichkeit der geometrischen Entkopplung beschrieben, wodurch der Speicheraufwand drastisch reduziert werden kann. RADMANOVIĆ und KATZ stellen 2010 einen optimierten Algorithmus vor, der mit zwei unterschiedlichen Zeitschrittlängen im Nah- und Fernfeld arbeitet [128].

Da trotz der Optimierungsansätze die benötigte Rechenzeit für großskalige bzw. komplexe Fragestellungen oft nicht akzeptabel ist und der begrenzte Arbeitsspeicher heutiger Computer ebenfalls Probleme bereitet, wird nach Alternativen der programmtechnischen Umsetzung gesucht. Eine Effizienzsteigerung ist mit verteilten Systemen möglich. Dabei wird das zu lösende Problem programmtechnisch aufgeteilt und gleichzeitig auf verschiedenen Systemen berechnet. Die Aufteilung erfordert zusätzliche Software, da neben dem Lösen des eigentlichen Problems auch die Kommunikation zwischen den Prozessen gesteuert werden muss. Einen parallelen FEM-SBFEM-Ansatz im Frequenzbereich stellen GENES und KOCAK [65] und GENES [64] vor. Eine Entwicklung für den Zeitbereich wird von SCHAUER et al. [143] diskutiert. Der Ansatz beinhaltet die Auswertung des Fernfeldes und die spätere Kopplung von FEM und SBFEM.

Die zahlreichen sehr unterschiedlichen Entwicklungen zeigen deutlich, dass ein erheblicher Bedarf an Verfahren zur Simulation randloser Gebiete besteht. Sie zeigen aber auch, dass eine optimale Umsetzung bisher nicht erarbeitet wurde. Einfache numerische Simulationen lassen sich mit den eingangs angesprochenen, relativ einfachen Randbedingungen realisieren. Steigt jedoch die Komplexität der Modelle, wird in der Regel gleichzeitig eine höhere Genauigkeit der Lösung verlangt, was die Erfüllung der Sommerfeld'schen Abstrahlbedingung erfordert. In diesem Fall steigt die Komplexität des numerischen Verfahrens und damit der Berechnungsaufwand sehr stark an. Bei der Untersuchung praxisrelevanter Fragestellungen liefern die derzeit eingesetzten Verfahren oftmals keine zufriedenstellenden Ergebnisse, sodass erheblicher Forschungsbedarf für effizientere numerische Verfahren vorhanden ist.

1.2. Ziel der Arbeit

In dieser Arbeit wird untersucht wie die Entwicklung eines effizienten gekoppelten Verfahrens, bestehend aus Finite Elemente Methode und *Scaled Boundary Element Method*, zur numerischen Simulation der Boden-Bauwerk-Interaktion erfolgen kann. Es soll untersucht werden, inwieweit dieser Ansatz für die Simulation großskaliger und damit realitätsnaher Modelle geeignet ist und nicht nur zur Berechnung akademischer Beispiele genutzt werden kann.

Die FEM eignet sich hervorragend dazu, das Nahfeld zu diskretisieren (vgl. Bild 1.2), da es mit ihr möglich ist, sehr komplexe Geometrien zu beschreiben. Hinzu kommt, dass die Berücksichtigung verschiedener Materialien und Materialeigenschaften auf Element- oder Knotenbasis

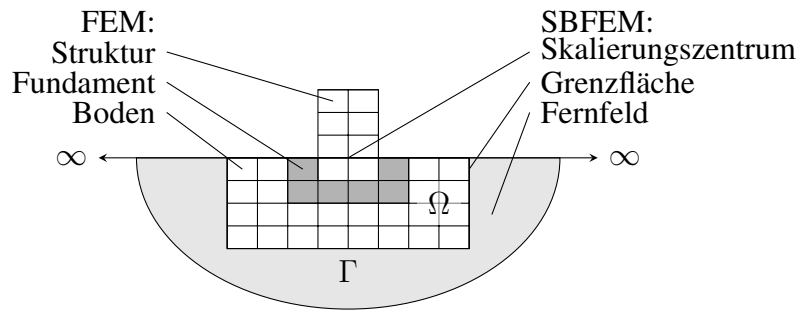


Bild 1.2.: Das Nahfeld Ω besteht beispielsweise aus Struktur, Fundament und Boden und wird mit der Finite Elemente Methode diskretisiert. Das angrenzende Fernfeld wird mit der *Scaled Boundary Finite Element Method* diskretisiert. Die Kopplung erfolgt über die gemeinsame Grenzfläche Γ .

realisierbar ist und die resultierenden Systemmatrizen bei entsprechender Formulierung symmetrisch sind, was sich positiv auf den Lösungsaufwand der Gleichungssysteme auswirkt.

Mit Hilfe der SBFEM hingegen lässt sich der unendliche Halbraum beschreiben (vgl. Bild 1.2). Wie bereits erwähnt, erfüllt sie ebenso wie die BEM die Sommerfeld'sche-Abstrahlbedingung implizit und exakt [21, 161]. Zusätzlich wird die räumliche Beschreibung um eine Dimension reduziert, da lediglich der Gebietsrand diskretisiert wird. Die Systemmatrizen der SBFEM sind wie bei der FEM symmetrisch und können mit den Freiheitsgraden der FEM auf dem gemeinsamen Rand Γ kombiniert werden, ohne weitere Freiwerte einzuführen. Dadurch entspricht die Größe des resultierenden, gekoppelten Gleichungssystems der Systemgröße des Nahfeldes. Ein weiterer nicht zu vernachlässigender Vorteil ist, dass die einmal berechneten Einflussmatrizen der SBFEM für unterschiedliche Konfigurationen des Nahfeldes verwendet werden können. Wird das Nahfeld hinsichtlich der Lasteinprägung oder der modellierten Struktur (Gebäude, Infrastruktur etc.) verändert, können die Einflussmatrizen erneut verwendet werden. Wird die Geometrie oder das Material des Kopplungsrandes geändert, sind die Einflussmatrizen jedoch neu zu bestimmen.

Eines haben alle im vorherigen Abschnitt vorgestellten Verfahren gemeinsam: Mit zunehmendem Detailgrad der numerischen Simulation steigen Berechnungsaufwand sowie Speicherbedarf. Mit steigendem Berechnungsaufwand steigt in der Regel gleichzeitig die benötigte Rechenzeit für die Lösung des gegebenen Problems. Dies gilt ebenfalls für den in dieser Arbeit verfolgten Lösungsansatz. Deshalb ist zu untersuchen, ob die sowohl rechenzeitintensiven als auch speicherintensiven numerischen Simulationen bei großskaligen und hoch aufgelösten Gebieten beherrschbar sind und Ergebnisse in angemessener Zeit generiert werden können.

Des Weiteren soll untersucht werden, welche Modellreduktionen herangezogen werden können, um den gekoppelten Ansatz aus Finiten Elemente Methode und *Scaled Boundary Finite Element Method* effizient für die eingangs beschriebenen Fragestellungen einzusetzen.

2. Theoretische Grundlagen

In diesem Kapitel werden die Modellgleichungen zusammengestellt, die zur Berechnung dreidimensionaler elastischer Körper erforderlich sind. Eine ausführliche Herleitung der Elastizitätstheorie erfolgt an dieser Stelle nicht und kann der Fachliteratur entnommen werden (vgl. z. B. [56, 72, 101, 111]). Begonnen wird mit der Beschreibung ideal elastischer Körper. Darauf folgt die Darstellung eines Stoffmodells für linear elastisches Materialverhalten sowie eine Zusammenfassung verschiedener Darstellungen von Spannungen im Raum. Daran anschließend werden die Beschreibung der Wellenausbreitung in einem ideal elastischen Medium sowie die Sommerfeld'sche Abstrahlbedingung diskutiert.

Die klassische Elastodynamik kann mit Hilfe der drei mechanischen Grundgleichungen Verzerrungszustand (Kinematik), Spannungszustand (Gleichgewicht) und Stoffgesetz beschrieben werden. Der Verzerrungszustand ist abhängig von den im System auftretenden Dehnungen und wird als Dehnungstensor dargestellt. Wird Isotropie vorausgesetzt und die Voigt'sche Notation verwendet, werden die 3×3 Einträge des räumlichen und symmetrischen Verzerrungstensors auf sechs Einträge in Vektorschreibweise reduziert [158]. Der Dehnungstensor

$$\boldsymbol{\epsilon} = [\epsilon_{11} \ \epsilon_{22} \ \epsilon_{33} \ 2\epsilon_{23} \ 2\epsilon_{31} \ 2\epsilon_{12}]^T \quad (2.1)$$

ist als Dehnungs-Verschiebungs-Beziehung

$$\boldsymbol{\epsilon} = \mathbf{D}\mathbf{u} \quad (2.2)$$

definiert und bildet das Produkt aus Operatormatrix

$$\mathbf{D} = \begin{bmatrix} \frac{\partial}{\partial x_1} & 0 & 0 \\ 0 & \frac{\partial}{\partial x_2} & 0 \\ 0 & 0 & \frac{\partial}{\partial x_3} \\ 0 & \frac{\partial}{\partial x_3} & \frac{\partial}{\partial x_2} \\ \frac{\partial}{\partial x_3} & 0 & \frac{\partial}{\partial x_1} \\ \frac{\partial}{\partial x_2} & \frac{\partial}{\partial x_1} & 0 \end{bmatrix} \quad (2.3)$$

und dem von den Raumkoordinaten abhängigen Verschiebungsvektor \mathbf{u} . Das Stoffmodell verknüpft Verzerrungstensor und Spannungstensor $\boldsymbol{\sigma}$. Die Spannungs-Dehnungs-Beziehung für elastisches Materialverhalten

$$\boldsymbol{\sigma} = \mathbf{E}\boldsymbol{\epsilon} \quad (2.4)$$

wird wie der Verzerrungstensor in der Voigt'schen Schreibweise dargestellt:

$$\boldsymbol{\sigma} = [\sigma_{11} \ \sigma_{22} \ \sigma_{33} \ \sigma_{23} \ \sigma_{31} \ \sigma_{12}]^T = [\sigma_{11} \ \sigma_{22} \ \sigma_{33} \ \tau_{23} \ \tau_{31} \ \tau_{12}]^T. \quad (2.5)$$

\mathbf{E} ist die Elastizitätsmatrix. Das Gleichgewicht eines Körpers kann mit

$$\mathbf{D}^T \boldsymbol{\sigma} + \mathbf{f} = 0 \quad (2.6)$$

beschrieben werden, wobei der erste Term die inneren und der zweite Term die äußeren Kräfte berücksichtigt. Für die Berücksichtigung des Schwingungsverhaltens des Tragwerks und der Wellenausbreitung im Fernfeld müssen die Massenträgheitskräfte $\rho \frac{d^2 \mathbf{u}(t)}{dt^2}$ sowie Dämpfungskräfte $\kappa \frac{d\mathbf{u}(t)}{dt}$ ergänzt werden,

$$\mathbf{D}^T \boldsymbol{\sigma}(t) + \mathbf{f}(t) - \rho \frac{d^2 \mathbf{u}(t)}{dt^2} - \kappa \frac{d\mathbf{u}(t)}{dt} = 0. \quad (2.7)$$

Die Bewegungsdifferentialgleichung beschreibt die Dynamik des Systems, Spannungstensor, Kraft- und Verschiebungsvektor sind zeitveränderlich. Die skalaren Größen ρ und κ berücksichtigen die Dichte beziehungsweise die Dämpfungseigenschaften des Kontinuums.

2.1. Berechnung und Darstellung von Spannungen

Die Beschreibung des Spannungsverhaltens kann auf unterschiedliche Art und Weise erfolgen. Eine mögliche Darstellung ist es, die koordinatenunabhängigen Invarianten des Spannungstensors zu verwenden:

$$I_1 = \sigma_{ii} = \sigma_{11} + \sigma_{22} + \sigma_{33}, \quad (2.8)$$

$$I_2 = \frac{1}{2} (\sigma_{ii} \sigma_{jj} - \sigma_{ij} \sigma_{ij}) = \sigma_{11} \sigma_{22} + \sigma_{22} \sigma_{33} + \sigma_{33} \sigma_{11} - (\sigma_{12}^2 + \sigma_{23}^2 + \sigma_{31}^2) \quad (2.9)$$

und

$$I_3 = |\sigma_{ij}| = \det \boldsymbol{\sigma} = \sigma_{11} \sigma_{22} \sigma_{33} + 2 \sigma_{12} \sigma_{31} \sigma_{23} - \sigma_{11} \sigma_{23}^2 - \sigma_{22} \sigma_{31}^2 - \sigma_{33} \sigma_{12}^2. \quad (2.10)$$

Somit ist der Spannungszustand unabhängig von Transformation und Rotation des Koordinatensystems. Mit der Determinanten des Spannungstensors

$$\sigma^3 - I_1 \sigma^2 - I_2 \sigma - I_3 = 0 \quad (2.11)$$

können die Hauptspannungen σ_i ermittelt werden. Ein allgemeiner Lösungsansatz für diese Art von Gleichungen ist beispielsweise [39] zu entnehmen. Ist der Spannungstensor in das Hauptachsensystem transformiert, sind sämtliche Schubspannungen gleich Null. Die Invarianten können dann in vereinfachter Form

$$I_1 = \sigma_{ii} = \sigma_1 + \sigma_2 + \sigma_3, \quad (2.12)$$

$$I_2 = -\sigma_1 \sigma_2 - \sigma_2 \sigma_3 - \sigma_3 \sigma_1 \quad (2.13)$$

und

$$I_3 = \sigma_1 \sigma_2 \sigma_3 \quad (2.14)$$

dargestellt werden. Die erste Invariante I_1 liefert die mittlere Normalspannung

$$\sigma_m = \frac{1}{3} I_1, \quad (2.15)$$

die in alle drei Raumrichtungen gleich groß ist. Mit Hilfe der mittleren Normalspannung wird der allgemeine Spannungstensor in einen hydrostatischen Anteil

$$\boldsymbol{\sigma}_m = \sigma_m \mathbf{I} = \begin{bmatrix} \sigma_m & & \\ & \sigma_m & \\ & & \sigma_m \end{bmatrix} \quad (2.16)$$

und einen deviatorischen Anteil

$$\mathbf{s} = \boldsymbol{\sigma} - \sigma_m \mathbf{I} = \begin{bmatrix} \sigma_{11} - \sigma_m & \sigma_{12} & \sigma_{31} \\ \sigma_{12} & \sigma_{22} - \sigma_m & \sigma_{23} \\ \sigma_{31} & \sigma_{23} & \sigma_{33} - \sigma_m \end{bmatrix} \quad (2.17)$$

zerlegt. Der hydrostatische Spannungstensor beschreibt den volumenändernden Anteil, der deviatorische den gestaltändernden Anteil der Spannungen. Auf Grundlage des deviatorischen Spannungstensors werden analog zu den Invarianten aus Gleichungen (2.8) bis (2.10) die deviatorischen Invarianten

$$J_1^D = s_{ii} = \sigma_{11} + \sigma_{22} + \sigma_{33} - 3\sigma_m, \quad (2.18)$$

$$J_2^D = \frac{1}{2} (s_{ij} s_{ij}) = \frac{1}{6} ((\sigma_{11} - \sigma_{22})^2 + (\sigma_{22} - \sigma_{33})^2 + (\sigma_{33} - \sigma_{11})^2) + \sigma_{12}^2 + \sigma_{23}^2 + \sigma_{31}^2 \quad (2.19)$$

und

$$J_3^D = |s_{ij}| = \det \mathbf{s} \quad (2.20)$$

berechnet.

2.2. Stoffmodelle

Das Stoffmodell, auch Stoffgesetz, Materialgesetz oder Werkstoffgesetz genannt, beschreibt den Zusammenhang von Dehnungen und Spannungen innerhalb eines Körpers und verknüpft in der mathematischen Darstellung den Spannungstensor mit dem Dehnungstensor, wie bereits in Gleichung (2.4) gezeigt. In dieser Arbeit wird der Terminus Stoffmodell genutzt, da es sich bei der Beschreibung der physikalischen Eigenschaften von Materialien eher um Modelle, die gewissen Gesetzmäßigkeiten folgen, als um allgemeingültige und belastbare Gesetze handelt. Das Verhalten unterschiedlicher Materialien oder auch Verbundmaterialien wird in der Regel durch rheologische Modelle im Stoffmodell abgebildet. Die rheologischen Modelle beschreiben das Materialverhalten mit Hilfe einfacher mechanischer Elemente wie Feder (Hooke'sches Element), Dämpfer (Newton'sches Element) und Reibklotz (St.-Venant'sches Element), die durch Parallel- oder Reihenschaltung miteinander kombiniert werden und so durch Schachtelung von Parallel- und Reihenschaltung beliebig komplexe Materialmodelle bilden können [89, 133].

Das einfachste anzunehmende Modell ist das Hooke'sche Stoffmodell, das isotropes, linear elastisches Materialverhalten abbildet. Das Material verformt sich aufgrund einwirkender Lasten, wobei Spannungen und Dehnungen in einem linearen Zusammenhang stehen, und kehrt bei vollständiger Entlastung in seinen Ausgangszustand zurück. Die Verformung ist somit vollkommen reversibel. Je nach Bedarf können wesentlich komplexere Stoffmodelle formuliert werden, die in der Lage sind viskoelastisches, elastisch-plastisches oder vollplastisches Verhalten abzubilden. Stoffmodelle, die viskoelastisches Materialverhalten abbilden, vereinen elastisches und viskoses Materialverhalten und kombinieren somit die Merkmale von Festkörper und Flüssigkeit. Modelle, die elastisch-plastisches Materialverhalten abbilden, zerlegen die Dehnung eines Körpers in einen reversiblen elastischen und einen irreversiblen plastischen Anteil. Bei vollplastischem Material ruft jede Art der Belastung eines Körpers eine irreversible Verformung hervor.

Des Weiteren ist es sinnvoll orthotropes sowie anisotropes Materialverhalten darzustellen. Isotrope Stoffmodelle beschreiben Werkstoffe, die sich in alle Raumrichtungen gleich verhalten und somit richtungsunabhängig sind. Ist dies nicht gewährleistet, wird von anisotropen Werkstoffen gesprochen. Anisotrope Werkstoffe besitzen unterschiedliche Eigenschaften in den unterschiedlichen Raumrichtungen und sind somit richtungsabhängig. Ein Spezialfall der Anisotropie ist die Orthotropie, bei der das Materialverhalten je nach Raumachse variiert, diese allerdings rechtwinklig aufeinander stehen und somit unabhängig voneinander sind. Je nach Stoffmodell unterscheidet sich der Aufbau der Elastizitätsmatrix. Im Falle eines beliebig anisotropen Materials verfügt die Elastizitätsmatrix über 21 elastische Konstanten. Bei der Beschreibung orthotroper Stoffmodelle ist die Anzahl auf neun elastische Konstanten reduziert. Verhält sich das Material in alle drei Raumrichtungen gleich, reichen zwei Konstanten zur Beschreibung des Materialverhaltens aus.

Vertiefende Informationen bezüglich des Werkstoffverhaltens unterschiedlicher Materialien sowie deren mathematische Beschreibung sind in zahlreichen Lehrbüchern wie beispielsweise denen von BARGEL und SCHULTZE [10], BRIKERT et al. [38], GROSS et al. [71] oder auch HORNBÖGEN et al. [78] gegeben. In Abhängigkeit von Anwendung und Materialeigenschaften ist das passende Stoffmodell zu wählen. Zur Beschreibung des Verformungsverhaltens von Tragwerken und Böden existiert eine Vielzahl an Stoffmodellen.

In dieser Arbeit werden alle nachfolgenden numerischen Simulationen unter Berücksichtigung eines linear elastischen Stoffmodells durchgeführt. Es wird vorausgesetzt, dass im Weiteren nicht berücksichtigte nicht-lineare Effekte lediglich im Nahfeld und nicht am Kopplungsrand auftreten dürfen und daher keinen Einfluss auf den zu untersuchenden effizienten gekoppelten FEM-SBFEM Ansatz haben.

Die Darstellung der Elastizitätsmatrix linear elastischen Materialienverhaltens

$$\mathbf{E} = \begin{bmatrix} 2\mu + \lambda & \lambda & \lambda & & & \\ \lambda & 2\mu + \lambda & \lambda & & & \\ \lambda & \lambda & 2\mu + \lambda & & & \\ & & & 2\mu & & \\ & & & & 2\mu & \\ & & & & & 2\mu \end{bmatrix} \quad (2.21)$$

erfolgt in der Regel mit Verwendung der Lamé'schen Konstanten

$$\mu = \frac{E}{2(1 + \nu)} \quad (2.22)$$

und

$$\lambda = \frac{\nu E}{(1 + \nu)(1 - 2\nu)}. \quad (2.23)$$

Ein isotropes, linear elastisches Materialverhalten wird somit mit den zwei Materialkenngrößen Elastizitätsmodul E und Poissonzahl oder auch Querdehnzahl ν beschrieben, wobei die Poissonzahl für den Wertebereich $0 \leq \nu < 0,5$ definiert ist. Alternativ ist diese Formulierung in Abhängigkeit des Kompressionsmoduls

$$K = \frac{E}{3 - 6\nu} \quad (2.24)$$

und des Schubmoduls

$$G = \frac{E}{2(1 + \nu)} \quad (2.25)$$

möglich. Die Lamé'schen Konstanten sind dann wie folgt anzupassen

$$\mu = G \quad \text{und} \quad \lambda = K - \frac{2G}{3}, \quad (2.26)$$

bevor sie zur Bestimmung der Elastizitätsmatrix aus Gleichung (2.21) herangezogen werden. Für die Beschreibung nicht-linearer Stoffmodelle sind diese beiden Kenngrößen nicht ausreichend. Je nach Materialmodell werden unterschiedlich viele zusätzliche Kenngrößen benötigt. Dabei gilt in der Regel, je komplexer das zu beschreibende Material, desto mehr Materialparameter sind erforderlich. Die Materialparameter müssen zuverlässig ermittelt werden, damit sichergestellt ist, dass das Modell die Realität entsprechend gut abbildet.

2.3. Wellenausbreitung

Bei der Betrachtung der Dynamik der Boden-Bauwerk-Interaktion ist ein wesentlicher Aspekt die Ausbreitung von Wellen im Boden. Dabei lassen sich unterschiedliche Wellentypen beobachten, die allgemein mit der Bewegungsgleichung

$$\rho \ddot{\mathbf{u}} = \mathbf{f} + (\lambda + 2\mu) \nabla (\nabla \cdot \mathbf{u}) - \mu \nabla \times (\nabla \times \mathbf{u}) \quad (2.27)$$

beschrieben werden können. Nachfolgend werden die wichtigsten Wellentypen kurz beschrieben, eine ausführliche Diskussion ist MESKOURIS et al. [115] zu entnehmen.

In einem homogenen isotropen ideal elastischen Raum breitet sich um den Punkt der Anregung eine kugelförmige Wellenfront aus. In radialer Richtung läuft die Kompressions- oder Longitudinalwelle, senkrecht dazu die Transversalwelle, die auch Schub- oder Scherwelle genannt wird. Bei diesen beiden Wellentypen handelt es sich um Raumwellen. Die Ausbreitungsgeschwindigkeit der Kompressionswelle c_P kann zu

$$c_P = \sqrt{\frac{E(1-\nu)}{\rho(1+\nu)(1-2\nu)}} = \sqrt{\frac{\lambda+2\mu}{\rho}} \quad (2.28)$$

bestimmt werden. Senkrecht zur Kompressionswelle breiten sich Transversalwellen c_S mit der Geschwindigkeit

$$c_S = \sqrt{\frac{E}{\rho(2+2\nu)}} = \sqrt{\frac{\mu}{\rho}} \quad (2.29)$$

aus. Da die Kompressionswelle deutlich schneller als die Transversalwelle ist und somit in der Regel die erste ist, die messtechnisch erfasst wird, wird diese auch Primär- oder P-Welle genannt. Analog dazu wird die zweite eintreffende Welle mit Sekundär- oder S-Welle bezeichnet. Aus den Gleichungen (2.28) und (2.29) ist leicht ersichtlich, dass mit zunehmender Steifigkeit des Materials die Wellengeschwindigkeit zunimmt, bei zunehmender Masse die Wellengeschwindigkeit abnimmt. Bild 2.1 zeigt das Verhältnis der beiden Wellengeschwindigkeiten zueinander in Abhängigkeit der Poissonzahl im Intervall von $0 \leq \nu < 0,5$. Diese Darstellung ist unabhängig von Elastizitätsmodul und Dichte und somit für alle homogenen isotropen ideal elastischen Materialien innerhalb des definierten Intervalls gültig.

Treffen Raumwellen auf eine freie Oberfläche, wie es beim unendlichen Halbraum der Fall ist, treten zusätzlich Oberflächenwellen auf. Diese Oberflächenwellen entstehen aufgrund der vorherrschenden Randbedingungen. Die Oberfläche ist spannungsfrei und die Spannung senkrecht zur Oberfläche gleich null, selbiges gilt für die dazugehörigen Schubspannungen. Aus einer Überlagerung der P-Welle und der vertikalen S-Welle entsteht die Rayleigh-Welle (R-Welle) [153]. Die Geschwindigkeit c_R der R-Welle wird nach RICHART et al. mit

$$\left(2p^2 - \frac{1}{c_S^2}\right)^2 - 4p^2 \sqrt{p^2 - \frac{1}{c_P^2}} \sqrt{p^2 - \frac{1}{c_S^2}} = 0 \quad (2.30)$$

bestimmt [131]. Mit $c_R = p^{-1}$ und unter der Annahme, dass $c_R < c_S$ ist, lässt sich Gleichung (2.30) eindeutig lösen und die Rayleigh-Wellengeschwindigkeit c_R ermitteln. Werden c_R und c_S ins Verhältnis zueinander gesetzt und ebenfalls über die Querdehnzahl $0 \leq \nu < 0,5$ aufgetragen, wie in Bild 2.2 gezeigt, dann folgt daraus $0,874 \leq c_R/c_S \leq 0,954$. Damit beträgt die Rayleigh-Wellengeschwindigkeit je nach Querdehnzahl zwischen 87,4% und 95,4% der Transversalwellengeschwindigkeit c_S .

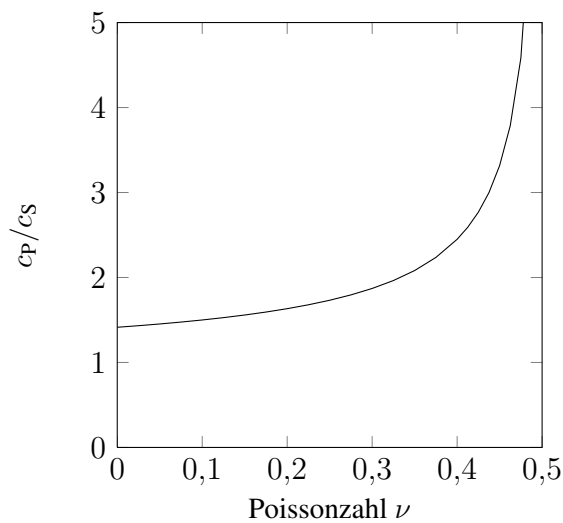


Bild 2.1.: Geschwindigkeitsverhältnis von Kompressions- und Transversalwelle in Abhängigkeit der Poissonzahl.

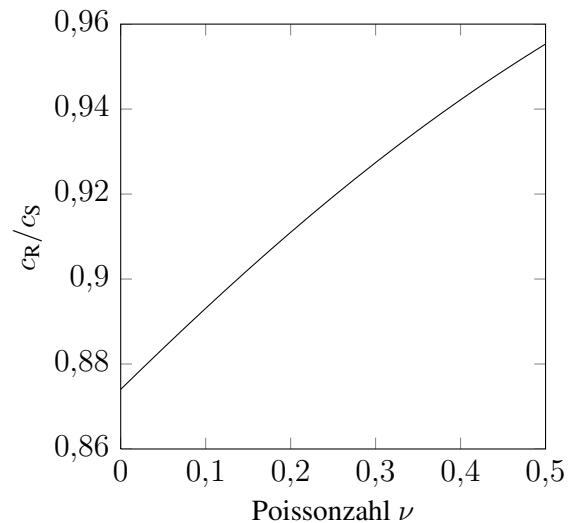


Bild 2.2.: Geschwindigkeitsverhältnis von Rayleigh- und Transversalwelle in Abhängigkeit der Poissonzahl.

Handelt es sich bei dem Halbraum nicht um einen homogenen, sondern einen geschichteten Halbraum, bestehend aus zwei oder mehr Schichten unterschiedlichen Materials, entsteht zusätzlich zur R-Welle eine Love-Welle (L-Welle). Die Love-Welle setzt sich aus horizontalen S-Wellen zusammen, die zwischen Oberfläche und Grenzschicht der unterschiedlichen Materialien reflektiert wird. Die Wellengeschwindigkeit der Love-Welle c_L ist abhängig von den Materialien der unterschiedlichen Schichten sowie deren Dicke. Eine analytische Berechnung ist nicht möglich, da keine eindeutige Lösung existiert.

2.4. Sommerfeld'sche Abstrahlbedingung

Die Sommerfeld'sche Abstrahlbedingung beschreibt das asymptotische Verhalten der Wellenausbreitung im randlosen bzw. unendlichen Raum und ist für unterschiedliche physikalische Phänomene wie beispielsweise Akustik, Elastodynamik und Elektromagnetismus gültig [144].

Wie im vorangegangenen Kapitel beschrieben, treten in der Elastodynamik verschiedene Wellentypen sowie deren Überlagerungen auf. An dieser Stelle erfolgt die Erläuterung der Abstrahlbedingung anhand eines vereinfachenden zeitharmonischen Ansatzes

$$u = \hat{u}e^{i\omega t} \quad (2.31)$$

für die Lösung der Wellengleichung (2.27), bei dem lediglich die Kompressionswelle berücksichtigt wird. Die Schubwellen werden hierbei außer acht gelassen, wodurch in diesem Ansatz nur noch eine Wellengeschwindigkeit auftritt. Für die anderen Wellentypen gilt dieser Ansatz analog [162]. Mathematisch lässt sich die Wellenausbreitung in ihrer einfachsten Form als Randwertproblem im Außenraum für Kompressionswellen im eingeschwungenen Zustand mit

der partiellen Differentialgleichung

$$\Delta \hat{\mathbf{u}} + k^2 \hat{\mathbf{u}} = 0 \quad (2.32)$$

formulieren. Werden kartesische Koordinaten vorausgesetzt, beschreibt $\hat{\mathbf{u}}(x, y, z)$ die räumliche Form der Welle. Der Laplace-Operator ist folglich durch $\Delta = \nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ definiert. Die Wellenzahl k bildet den Quotienten aus Eigenfrequenz ω und Wellengeschwindigkeit c .

Wie bereits eingangs erwähnt, beschreibt die Sommerfeld'sche Abstrahlbedingung

$$\lim_{r \rightarrow \infty} r \left(\frac{\partial \hat{\mathbf{u}}}{\partial r} - ik \hat{\mathbf{u}} \right) = 0 \quad (2.33)$$

mit $k = \frac{\omega}{c_p}$ und $r = \sqrt{x^2 + y^2 + z^2}$ das asymptotische Verhalten der Welle [148]. Es wird gefordert, dass die Amplitude der Welle mit zunehmendem Abstand r gegen null strebt. Die Rücktransformation dieses Ansatzes in den Zeitbereich ist durchführbar, indem die zeitliche Ableitung des zeit-harmonischen Ansatzes (2.31) gebildet wird

$$\frac{\partial u}{\partial t} = \hat{u} i \omega e^{i \omega t} \quad (2.34)$$

und in Gleichung (2.33) eingesetzt wird. Daraus folgt die Darstellung der Abstrahlbedingung im Zeitbereich

$$\lim_{r \rightarrow \infty} r \left(\frac{\partial u}{\partial r} - \frac{1}{c} \frac{\partial u}{\partial t} \right) = 0. \quad (2.35)$$

Eine ausführliche Diskussion der Sommerfeld'schen Abstrahlbedingung ist unter anderem in [148, 162] zu finden.

3. Numerische Modellbildung

Nachfolgend werden die zur Simulation von Nah- und Fernfeld verwendeten Simulationsverfahren sowie die Kopplung dieser beiden Verfahren diskutiert. Die hier beschriebene Herleitung der numerischen Verfahren basiert auf den zuvor in Kapitel 2 diskutierten theoretischen sowie mechanischen Grundlagen. Hierzu wird zunächst eine kurze Einführung zur Finite Elemente Methode im Zeitbereich gegeben, die zur Diskretisierung des Nahfeldes eingesetzt wird. Für weiterführende, vertiefende Erörterungen wird an dieser Stelle exemplarisch auf die Fachliteratur von BATHE [14] und ZIENKEWICZ et al. [170, 171] verwiesen. Anschließend werden die Grundlagen der *Scaled Boundary* Finite Elemente Methode zur Simulation des Fernfeldes diskutiert.

Zur besseren Verständlichkeit wird für die Darstellung der beiden numerischen Methoden eine einheitliche Notation für die Finite Elemente Methode als auch *Scaled Boundary* Finite Elemente Methode gewählt. Dies führt dazu, dass die Lesart der *Scaled Boundary* Finite Elemente Methode von der ursprünglichen Art und Weise, wie sie von WOLF und SONG [162] eingeführt wurde, stark abweicht.

3.1. Finite Elemente Methode

Differentialgleichungen können auf verschiedene Weise mit Hilfe der Finite Elemente Methode gelöst werden. Um Fragestellungen der Stukturdynamik zu analysieren, bietet sich eine Formulierung in der schwachen Form auf Basis virtueller Arbeiten an. Die Gleichung (2.7) wird zu diesem Zweck entsprechend umgeformt. Virtuelle Größen sind im Weiteren mit δ gekennzeichnet. Hierbei werden sowohl innere als auch äußere Arbeiten in einem definierten Gebiet Ω und auf dessen Gebietsrand Γ betrachtet und summiert

$$-\delta W = \int_{\Omega} \delta \epsilon^T \boldsymbol{\sigma} d\Omega + \int_{\Omega} \delta \mathbf{u}^T \left(\kappa \frac{d\mathbf{u}}{dt} + \rho \frac{d^2 \mathbf{u}}{dt^2} \right) d\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{f} d\Omega - \int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} d\Gamma = 0. \quad (3.1)$$

Die vektoriellen Größen \mathbf{u} , $\frac{d\mathbf{u}}{dt} = \dot{\mathbf{u}}$ und $\frac{d^2 \mathbf{u}}{dt^2} = \ddot{\mathbf{u}}$ repräsentieren Verschiebung, Geschwindigkeit und Beschleunigung. Einwirkende Kräfte bzw. Spannungen sind mit \mathbf{f} bzw. \mathbf{t} dargestellt. Die Parameter κ und ρ beschreiben Dämpfungsgrad und Dichte des Systems. In die aufgestellte Arbeitsintegralgleichung (3.1) werden die kinematischen Beziehungen sowie das Materialmodell aus Abschnitt 2.2, Gleichung (2.2) und (2.4), eingesetzt. Da die Dehnung $\delta \epsilon$ virtuell ist, ist für die Spannungs-Dehnungs-Beziehung ebenfalls eine virtuelle Verschiebung $\delta \mathbf{u}$ zu berücksichtigen, woraus sich die folgende Gleichung ergibt:

$$-\delta W = \int_{\Omega} (\mathbf{D} \delta \mathbf{u})^T \mathbf{E} \mathbf{D} \mathbf{u} d\Omega + \int_{\Omega} \delta \mathbf{u}^T (\kappa \dot{\mathbf{u}} + \rho \ddot{\mathbf{u}}) d\Omega - \int_{\Omega} \delta \mathbf{u}^T \mathbf{f} d\Omega - \int_{\Gamma} \delta \mathbf{u}^T \mathbf{t} d\Gamma = 0. \quad (3.2)$$

Sobald die Verschiebungen \mathbf{u} die geometrischen Randbedingungen erfüllen, entspricht Gleichung (3.2) der Gleichgewichtsbedingung. Zur Bestimmung der unbekannten Verschiebungen

\mathbf{u} werden die Integrale aus Gleichung (3.2) in eine finite Anzahl Teilintegrale δW aufgeteilt und summiert, sodass

$$\delta W = \sum_{i=1}^n \delta W_i = 0 \quad (3.3)$$

gilt. Diese Teilintegrale werden dann von einer endlichen Zahl finiter Elemente repräsentiert und mit Hilfe definierter Ansatzfunktionen $N(\xi_1, \xi_2, \xi_3)$ ausgewertet. Dabei werden die approximierten Verschiebungen \mathbf{u} elementweise bestimmt

$$\mathbf{u}(\mathbf{x}) = \sum_{i=1}^m N^i(\boldsymbol{\xi}) \mathbf{u}^i = \mathbf{N} \mathbf{u}. \quad (3.4)$$

Eine einfache Auswertung der Ansatzfunktionen wird ermöglicht, indem die globalen Koordinaten \mathbf{x} auf Elementebene in lokale Koordinaten $\boldsymbol{\xi}$ transformiert werden. Für die virtuelle Verschiebung ergibt sich analog

$$\delta \mathbf{u}(\mathbf{x}) = \mathbf{N} \delta \mathbf{u}. \quad (3.5)$$

Dieses Vorgehen sorgt im weiteren Verlauf für symmetrische Systemmatrizen. Einsetzen der Gleichungen (3.4) and (3.5) in Gleichung (3.2) liefert

$$\begin{aligned} -\delta W &= \delta \mathbf{u}^T \int_{\Omega} (\mathbf{D}\mathbf{N})^T \mathbf{E}(\mathbf{D}\mathbf{N}) \mathbf{u} \, d\Omega \\ &\quad + \delta \mathbf{u}^T \int_{\Omega} \mathbf{N}^T (\kappa \mathbf{N} \dot{\mathbf{u}} + \rho \mathbf{N} \ddot{\mathbf{u}}) \, d\Omega \\ &\quad - \delta \mathbf{u}^T \int_{\Omega} \mathbf{N}^T \mathbf{f} \, d\Omega \\ &\quad - \delta \mathbf{u}^T \int_{\Gamma} \mathbf{N}^T \mathbf{t} \, d\Gamma = 0. \end{aligned} \quad (3.6)$$

Dieser wird mit

$$\mathbf{M} = \int_{\Omega} \mathbf{N}^T (\rho \mathbf{N}) \, d\Omega, \quad (3.7)$$

$$\mathbf{C} = \int_{\Omega} \mathbf{N}^T (\kappa \mathbf{N}) \, d\Omega, \quad (3.8)$$

$$\mathbf{K} = \int_{\Omega} (\mathbf{D}\mathbf{N})^T \mathbf{E}(\mathbf{D}\mathbf{N}) \, d\Omega, \quad (3.9)$$

$$\mathbf{p} = \int_{\Omega} \mathbf{N}^T \mathbf{f} \, d\Omega + \int_{\Gamma} \mathbf{N}^T \mathbf{t} \, d\Gamma \quad (3.10)$$

in die Bewegungsdifferentialgleichung

$$\mathbf{M} \ddot{\mathbf{u}} + \mathbf{C} \dot{\mathbf{u}} + \mathbf{K} \mathbf{u} = \mathbf{p} \quad (3.11)$$

in Matrix-Vektor-Schreibweise überführt. Hier ist \mathbf{M} die Massenmatrix, \mathbf{C} die Dämpfungsmatrix und \mathbf{K} die Steifigkeitsmatrix. Auf der rechten Seite der Gleichung steht der Lastvektor \mathbf{p} . Die Ermittlung des frequenzabhängigen Dämpfungsparameters κ wird in dieser Grundsatzuntersuchung nicht betrachtet, deshalb wird hier die Dämpfungsmatrix als Rayleigh-Dämpfung umgesetzt [152]. Dabei wird die Dämpfungsmatrix aus einer gewichteten Kombination von Massen- und Steifigkeitsmatrix

$$\mathbf{C} = c_m \mathbf{M} + c_k \mathbf{K} \quad (3.12)$$

bestimmt. Das Problem, die skalaren Wichtungsfaktoren c_m und c_k angemessen zu wählen, besteht allerdings weiterhin.

Da die Beschreibung der Dämpfung in der Diskussion der Grundlagen unberücksichtigt blieb, soll diese an dieser Stelle der Vollständigkeit halber kurz aufgegriffen werden. Es sei darauf hingewiesen, dass die Beschreibung der Dämpfung Teil der mechanischen Beschreibung und nicht Finite Elemente Methode spezifisch ist. Eine Möglichkeit zur Bestimmung der Wichtungsfaktoren liefert die nachfolgende Gleichung

$$\varsigma_i = \frac{c_m}{2\omega_i} + \frac{c_k \omega_i}{2}, \quad (3.13)$$

diese wird als modale Dämpfung bezeichnet und ist definiert als Summe der masseproportionalen $\frac{c_m}{2\omega_i}$ sowie steifigkeitsproportionalen $\frac{c_k \omega_i}{2}$ Dämpfung. Eine Modalanalyse liefert alle Eigenfrequenzen ω_i des zu untersuchenden Systems. Jeder Eigenfrequenz ω_i ist ein Dämpfungsgrad ς_i zuzuordnen. PETERSEN empfiehlt, höheren Frequenzen einen größeren Dämpfungsgrad zuzuweisen [122] und somit Oberschwingungen verstärkt zu dämpfen, da Systeme primär dazu neigen, in der Grundschwingung zu schwingen. Für $i > 2$ folgt aus Gleichung (3.13) ein überbestimmtes Gleichungssystem. Wie dieser theoretische Ansatz mit realen Messungen in Einklang gebracht werden kann, ist noch nicht abschließend geklärt. An dieser Stelle besteht nach wie vor Forschungsbedarf [122].

Alternativ ist der nachfolgend beschriebene Ansatz zur Abschätzung der Wichtungsfaktoren zulässig. Davon ausgehend, dass zur eindeutigen Beschreibung von c_m und c_k $i = 2$ sein muss und der Dämpfungsgrad über alle Frequenzen als konstant angenommen ($\varsigma_i = \varsigma$) wird, kann c_m mit

$$c_m = \varsigma \frac{2\omega_1 \omega_2}{\omega_1 + \omega_2} \quad (3.14)$$

und c_k mit

$$c_k = \varsigma \frac{2}{\omega_1 + \omega_2} \quad (3.15)$$

bestimmt werden. Somit sind die für die Dämpfung benötigten Größen von lediglich zwei Eigenfrequenzen sowie dem Dämpfungsgrad abhängig. Näherungsweise kann der Dämpfungsgrad, wie beispielsweise von PETERSEN ausführlich beschrieben [122], ermittelt werden. Für ω_1 und ω_2 wird empfohlen, die ersten zwei Grundschwingungen des Systems heranzuziehen.

3.2. Zeitschrittintegrationsverfahren

Die in Gleichung (3.11) aufgestellte Bewegungsdifferentialgleichung wird mit Hilfe impliziter Einschrittverfahren gelöst. Im Gegensatz zu expliziten Verfahren, die zur Bestimmung der neuen Werte lediglich auf bereits bekannte Größen zurückgreifen, bestimmen implizite Zeitschrittintegrationsverfahren die Werte des neuen Zeitschritts anhand des vorangegangenen sowie des aktuellen Zeitschritts. Die in dieser Arbeit nachfolgend diskutierten Verfahren gehören alle zur Familie der Newmark-Verfahren. Bei allen betrachteten Problemstellungen wird davon ausgegangen, dass die Anfangswertbedingungen zum Zeitpunkt $t_0 := t = 0$

$$\mathbf{u}(t_0) = \mathbf{u}_0 \quad \text{und} \quad (3.16)$$

$$\dot{\mathbf{u}}(t_0) = \dot{\mathbf{u}}_0 \quad (3.17)$$

bekannt sind und die zu untersuchende zeitliche Periode T in n Zeitschritte der konstanten Größe $\Delta t = \frac{T}{n}$ zerlegt werden kann. Die Zeitschrittlänge Δt beschreibt somit den Abstand zwischen zwei Zeitschritten t_n und t_{n+1} , wodurch der aktuelle Zeitschritt t_{n+1} durch $t_{n+1} = t_n + \Delta t$ definiert ist.

Im Folgenden werden die verschiedenen Zeitschrittintegrationsverfahren, wie sie hier in dieser Arbeit verwendet werden, vorgestellt. Dies soll einen Überblick über die Vor- und Nachteile der einzelnen Verfahren geben und somit verdeutlichen, wann welches Verfahren am besten geeignet ist.

3.2.1. Newmark-Verfahren

NEWMARK stellte 1959 eine Methode zur strukturdynamischen Berechnung vor [120]. Sind die Werte der Orts-, Geschwindigkeits- und Beschleunigungsvektoren zum Zeitschritt t_n bekannt, können die neuen Werte für den Zeitschritt t_{n+1} mit

$$\mathbf{u}_{n+1} = \mathbf{u}_n + \Delta t \dot{\mathbf{u}}_n + \left(\frac{1}{2} - \beta \right) \Delta t^2 \ddot{\mathbf{u}}_n + \beta \Delta t^2 \ddot{\mathbf{u}}_{n+1}, \quad (3.18)$$

$$\dot{\mathbf{u}}_{n+1} = \dot{\mathbf{u}}_n + (1 - \gamma) \Delta t \ddot{\mathbf{u}}_n + \gamma \Delta t \ddot{\mathbf{u}}_{n+1} \quad (3.19)$$

und

$$\mathbf{M} \ddot{\mathbf{u}}_{n+1} + \mathbf{C} \dot{\mathbf{u}}_{n+1} + \mathbf{K} \mathbf{u}_{n+1} = \mathbf{p}_{n+1} \quad (3.20)$$

bestimmt werden. Dass es sich, wie bereits erwähnt, um implizite Verfahren handelt, ist anhand der Gleichungen (3.18) und (3.19) deutlich zu sehen. Die neuen zu bestimmenden Größen des Zeitschritts t_{n+1} hängen sowohl von den Werten des Zeitschritts t_n als auch von Werten des

Schritts t_{n+1} ab. Über die Genauigkeit und Stabilität des Verfahrens entscheiden die Parameter β und γ . Werden diese Parameter wie folgt gewählt,

$$\gamma \geq \frac{1}{2} \quad \text{und} \quad (3.21)$$

$$\beta \geq \frac{1}{4} \left(\gamma + \frac{1}{2} \right)^2 \quad (3.22)$$

ist der Algorithmus unbedingt stabil [77]. Für $\gamma > \frac{1}{2}$ ist das Newmark-Verfahren numerisch dissipativ, somit kann bei Bedarf numerische Dämpfung hinzugefügt werden. Zum einen ermöglicht dies einen stabilen Algorithmus bei großen Zeitschrittlängen, zum anderen ist eine Dämpfung in der Strukturmechanik durchaus gewollt, da bei realen Fragestellungen in der Regel Dämpfung auftritt. Ein ungewollter Nebeneffekt ist allerdings, dass sämtliche Frequenzbereiche gedämpft und somit verfälscht werden. Dies hat zur Folge, dass die Amplituden aller auftretenden Frequenzen reduziert sind, wodurch es zu Amplitudenfehlern kommt.

3.2.2. Hilber-Hughes-Taylor- α -Verfahren

Das von HILBER, HUGHES und TAYLOR 1977 vorgestellte HHT- α -Verfahren wurde entwickelt, um die ungewollten Nebeneffekte des zuvor beschriebenen Newmark-Verfahrens zu minimieren [77]. Ziel ist es, lediglich hohe Frequenzen, die für numerische Instabilität sorgen können, zu dämpfen, niedrige und mittlere Frequenzen jedoch nicht zu verfälschen. Um dies zu ermöglichen wird in die Gleichung (3.20) ein weiterer Parameter α eingeführt:

$$\mathbf{M}\ddot{\mathbf{u}}_{n+1} + (1 + \alpha)\mathbf{C}\dot{\mathbf{u}}_{n+1} - \alpha\mathbf{C}\dot{\mathbf{u}}_n + (1 + \alpha)\mathbf{K}\mathbf{u}_{n+1} - \alpha\mathbf{K}\mathbf{u}_n = \mathbf{p}_{n+1} \quad (3.23)$$

Die Aktualisierungsregeln aus Gleichung (3.18) und (3.19) bleiben unverändert. Die Grenzen der Parameter werden von HILBER et al. [77] mit

$$-\frac{1}{2} \leq \alpha \leq 0, \quad (3.24)$$

$$\beta = \frac{(1 - \alpha)^2}{4} \quad \text{und} \quad (3.25)$$

$$\gamma = \frac{1}{2} - \alpha \quad (3.26)$$

angegeben.

3.2.3. Generalized- α -Verfahren

Das von CHUNG et al. 1993 vorgestellte Generalized- α -Verfahren [42] basiert auf einer Kombination des zuvor diskutierten HHT- α -Verfahrens sowie des von WOOD, BOSSAK und ZIEN-

KIEWICZ vorgestellten WBZ- α -Verfahrens [165]. Ziel dieses Verfahrens ist es, die hohen Frequenzen zu dämpfen, die niedrigen Frequenzen jedoch so wenig wie möglich zu beeinflussen. Beim Generalized- α -Verfahren wird Gleichung (3.20) in

$$(1 - \alpha_m)\mathbf{M}\ddot{\mathbf{u}}_{n+1} + \alpha_m\mathbf{M}\ddot{\mathbf{u}}_n + (1 - \alpha_f)\mathbf{C}\dot{\mathbf{u}}_{n+1} + \alpha_f\mathbf{C}\dot{\mathbf{u}}_n + (1 - \alpha_f)\mathbf{K}\mathbf{u}_{n+1} + \alpha_f\mathbf{K}\mathbf{u}_n = \mathbf{p}_{n+1} - \alpha_f\Delta t\mathbf{p}_n \quad (3.27)$$

überführt. Die Ausgangsgleichung wird um zwei α Parameter ergänzt, sodass die numerische Dissipation über die Parameter α_f und α_m vom Anwender besser gesteuert werden kann. Die Bestimmung der aktuellen Verschiebungen und Geschwindigkeiten erfolgt wie schon zuvor in Gleichung (3.18) und (3.19) beschrieben. Die das Zeitschrittverfahren regulierenden Parameter α_m , α_f , β und γ können innerhalb der vorgegebenen Grenzen

$$\alpha_m \leq \alpha_f \leq \frac{1}{2}, \quad (3.28)$$

$$\beta \geq \frac{1}{4} + \frac{1}{2}(\alpha_f - \alpha_m) \text{ und} \quad (3.29)$$

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f \quad (3.30)$$

an die Problemstellung angepasst werden, um den Algorithmus gegebenenfalls zu stabilisieren. Wie in [42] beschrieben, verfügt dieses Verfahren über einige Vorteile im Vergleich zu den beiden vorangegangenen Verfahren. Wird der Parameter α_m gleich Null gewählt, entspricht das Verfahren dem HHT- α -Verfahren. Wird zusätzlich auch der Parameter α_f gleich Null gewählt, folgt die Zeitschrittintegration dem klassischen Newmark-Verfahren.

3.2.4. Allgemeiner Lösungsansatz

Unabhängig vom gewählten Verfahren wird das Zeitschrittintegrationsverfahren gelöst, indem Gleichung (3.18) nach

$$\ddot{\mathbf{u}}_{n+1} = \frac{1}{\beta\Delta t^2} (\mathbf{u}_{n+1} - \mathbf{u}_n) - \frac{1}{\beta\Delta t} \dot{\mathbf{u}}_n + \left(1 - \frac{1}{2\beta}\right) \ddot{\mathbf{u}}_n \quad (3.31)$$

umgestellt wird und zusammen mit Gleichung (3.19) in die jeweilige Bewegungsdifferentialgleichung (3.20), (3.23) oder (3.27) eingesetzt wird. Dies sei für das Generalized- α -Verfahren exemplarisch gezeigt

$$\begin{aligned} & (1 - \alpha_m)\mathbf{M} \left(\frac{1}{\beta\Delta t^2} (\mathbf{u}_{n+1} - \mathbf{u}_n) - \frac{1}{\beta\Delta t} \dot{\mathbf{u}}_n + \left(1 - \frac{1}{2\beta}\right) \ddot{\mathbf{u}}_n \right) + \alpha_m\mathbf{M}\ddot{\mathbf{u}}_n + \\ & (1 - \alpha_f)\mathbf{C} \left(\frac{\gamma}{\beta\Delta t} (\mathbf{u}_{n+1} - \mathbf{u}_n) + \left(1 - \frac{\gamma}{\beta}\right) \dot{\mathbf{u}}_n + \left(\Delta t - \frac{\gamma\Delta t}{2\beta}\right) \ddot{\mathbf{u}}_n \right) + \alpha_f\mathbf{C}\dot{\mathbf{u}}_n + \\ & (1 - \alpha_f)\mathbf{K}\mathbf{u}_{n+1} + \alpha_f\mathbf{K}\mathbf{u}_n = \mathbf{p}_{n+1} - \alpha_f\Delta t\mathbf{p}_n. \end{aligned} \quad (3.32)$$

Durch Umstellen der Gleichung folgt die diskretisierte Bewegungsgleichung

$$\begin{aligned}
& \left((1 - \alpha_m) \frac{1}{\beta \Delta t^2} \mathbf{M} + (1 - \alpha_f) \frac{\gamma}{\beta \Delta t} \mathbf{C} + (1 - \alpha_f) \mathbf{K} \right) \mathbf{u}_{n+1} = \mathbf{p}_{n+1} - \alpha_f \Delta t \mathbf{p}_n \\
& + (1 - \alpha_m) \mathbf{M} \left(\frac{1}{\beta \Delta t^2} \mathbf{u}_n + \frac{1}{\beta \Delta t} \dot{\mathbf{u}}_n - \left(1 - \frac{1}{2\beta} \right) \ddot{\mathbf{u}}_n \right) - \alpha_m \mathbf{M} \ddot{\mathbf{u}}_n \\
& + (1 - \alpha_f) \mathbf{C} \left(\frac{\gamma}{\beta \Delta t} \mathbf{u}_n - \left(1 - \frac{\gamma}{\beta} \right) \dot{\mathbf{u}}_n - \left(\Delta t - \frac{\gamma \Delta t}{2\beta} \right) \ddot{\mathbf{u}}_n \right) - \alpha_f \mathbf{C} \dot{\mathbf{u}}_n + \alpha_f \mathbf{K} \mathbf{u}_n.
\end{aligned} \tag{3.33}$$

Werden die effektive Steifigkeitsmatrix mit

$$\hat{\mathbf{K}} = (1 - \alpha_m) \frac{1}{\beta \Delta t^2} \mathbf{M} + (1 - \alpha_f) \frac{\gamma}{\beta \Delta t} \mathbf{C} + (1 - \alpha_f) \mathbf{K} \tag{3.34}$$

und der effektive Lastvektor mit

$$\begin{aligned}
\hat{\mathbf{p}} &= \mathbf{p}_{n+1} - \alpha_f \Delta t \mathbf{p}_n \\
&+ (1 - \alpha_m) \mathbf{M} \left(\frac{1}{\beta \Delta t^2} \mathbf{u}_n + \frac{1}{\beta \Delta t} \dot{\mathbf{u}}_n + \left(\frac{1}{2\beta} - 1 \right) \ddot{\mathbf{u}}_n \right) - \alpha_m \mathbf{M} \ddot{\mathbf{u}}_n \\
&+ (1 - \alpha_f) \mathbf{C} \left(\frac{\gamma}{\beta \Delta t} \mathbf{u}_n + \left(\frac{\gamma}{\beta} - 1 \right) \dot{\mathbf{u}}_n + \left(\frac{\gamma \Delta t}{2\beta} - \Delta t \right) \ddot{\mathbf{u}}_n \right) - \alpha_f \mathbf{C} \dot{\mathbf{u}}_n + \alpha_f \mathbf{K} \mathbf{u}_n
\end{aligned} \tag{3.35}$$

definiert, ist in jedem Zeitschritt das Gleichungssystem

$$\hat{\mathbf{K}} \mathbf{u}_{n+1} = \hat{\mathbf{p}} \tag{3.36}$$

zu lösen. Das Newmark-Verfahren und das Hilber-Hughes-Taylor- α -Verfahren werden analog zum Generalized- α -Verfahren gelöst. Alternativ können α_m und α_f zu Null gesetzt werden. Damit entfallen einige Terme, sodass die effektive Steifigkeitsmatrix und der effektive Lastvektor für das Newmark-Verfahren mit

$$\hat{\mathbf{K}} = \frac{1}{\beta \Delta t^2} \mathbf{M} + \frac{\gamma}{\beta \Delta t} \mathbf{C} + \mathbf{K} \tag{3.37}$$

und

$$\begin{aligned}
\hat{\mathbf{p}} &= \mathbf{p}_{n+1} \\
&+ \mathbf{M} \left(\frac{1}{\beta \Delta t^2} \mathbf{u}_n + \frac{1}{\beta \Delta t} \dot{\mathbf{u}}_n + \left(\frac{1}{2\beta} - 1 \right) \ddot{\mathbf{u}}_n \right) \\
&+ \mathbf{C} \left(\frac{\gamma}{\beta \Delta t} \mathbf{u}_n + \left(\frac{\gamma}{\beta} - 1 \right) \dot{\mathbf{u}}_n + \left(\frac{\gamma \Delta t}{2\beta} - \Delta t \right) \ddot{\mathbf{u}}_n \right)
\end{aligned} \tag{3.38}$$

gegeben sind. Diese können ebenfalls zur Bestimmung des nächsten Zeitschritts in Gleichung (3.36) eingesetzt werden.

3.2.5. Parameterwahl

Die Wahl der Parameter gestaltet sich je nach Problemstellung beliebig komplex. CHUNG et al. schlagen vor, diese in Abhängigkeit des Dissipationsparameters p_∞ mit

$$0 \leq p_\infty \leq 1, \quad (3.39)$$

$$\alpha_m = \frac{2p_\infty - 1}{p_\infty + 1}, \quad (3.40)$$

$$\alpha_f = \frac{p_\infty}{p_\infty + 1}, \quad (3.41)$$

$$\beta = \frac{1}{4}(1 - \alpha_m + \alpha_f)^2 \text{ und} \quad (3.42)$$

$$\gamma = \frac{1}{2} - \alpha_m + \alpha_f \quad (3.43)$$

zu ermitteln [42].

Auf Grundlage der gegebenen Randbedingungen (Abschnitt 3.2.1 bis 3.2.3, Gleichung (3.21), (3.24) und (3.28)) lassen sich die Parameter der unterschiedlichen Zeitschrittintegrationsverfahren, wie in den Bildern 3.1a bis 3.1c gezeigt, bestimmen. Alle Parameter sind dort in Abhängigkeit des Parameters γ dargestellt, um die Vergleichbarkeit der einzelnen Parameter zu gewährleisten. Die in Tabelle 3.1 zusammengefassten Parameterkombinationen haben sich als geeignet erwiesen. Für weitere Details sei exemplarisch auf die Fachliteratur verwiesen [42, 77, 120, 165]. Eine genauere Betrachtung erfolgt bei der Diskussion der untersuchten Problemstellungen in Kapitel 5.

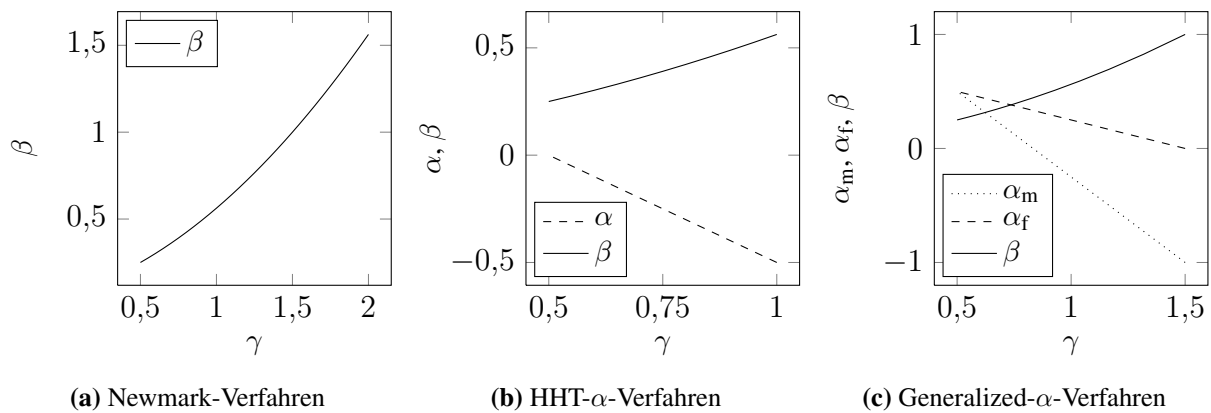


Bild 3.1.: Darstellung der Zeitintegrationsparameter des Newmark-Verfahrens **a**, des HHT- α -Verfahrens **b** sowie des Generalized- α -Verfahrens **c** in Abhängigkeit des Parameters γ .

Zeitschrittintegrationsverfahren	α_m	α_f	α	β	γ
Newmark ohne Dämpfung (Trapez-Regel)				0,25	0,5
Trapez-Regel mit α -Dämpfung (HHT- α)			-0,1	0,25	0,5
Newmark (mit Dämpfung)				0,3025	0,6
Newmark mit α -Dämpfung (HHT- α)			-0,1	0,3025	0,6
HHT- α			-0,25	0,390625	0,75
Generalized- α ($p_\infty = 0,6$)	0,125	0,375		0,390625	0,75

Tabelle 3.1.: Übersicht über die Parameterwahl der Zeitschrittintegrationsverfahren.

3.2.6. Zeitschrittlänge

Die Zeitschrittlänge Δt ist von zwei Größen maßgeblich abhängig. Zum einen von der Elementlänge ℓ_{el} und damit von der Diskretisierung der Geometrie. Je kleiner die Elementlänge ℓ_{el} desto höher ist die Auflösung des Modells und um so genauer werden die Ergebnisse der Berechnung. Zum anderen von der Wellengeschwindigkeit c_p (vgl. Abschnitt 2.3) und somit unmittelbar vom verwendeten Material. Die Zeitschrittlänge lässt sich mit der kritischen Zeitschrittlänge

$$\Delta t \leq \Delta t_{krit} = \frac{\ell_{el}}{n c_p} \quad (3.44)$$

abschätzen. Damit sichergestellt ist, dass die schnellste Welle im System nicht innerhalb eines Zeitschritts ein oder gar mehrere Elemente durchläuft, wird $n > 1$ gefordert.

Auf dieser Grundlage können ebenfalls die höchste auflösbare Frequenz

$$f_{max} = \frac{1}{\Delta t} = \frac{n c_p}{\ell_{el}} \quad (3.45)$$

sowie die kleinste auflösbare Wellenlänge

$$\lambda_{max} = \frac{c_p}{f_{max}} = n \ell_{el} \quad (3.46)$$

bestimmt werden. Eine Sinus-Schwingung der Frequenz f wird an n Punkten erfasst. Ist die Anzahl der Punkte im Intervall $6 \leq n \leq 10$ gewählt, reicht dies in der Regel aus, die Schwingung hinreichend genau abzubilden. Sollte dies nicht der gewünschten Genauigkeit entsprechen, ist die Anzahl der Punkte zu erhöhen und die Zeitschrittlänge entsprechend anzupassen.

3.2.7. Energiebilanz

Die Betrachtung der Energiebilanz eignet sich sehr gut, um die Genauigkeit eines Zeitintegrationsverfahrens zu testen. Bei einem ungedämpften dynamischen System mit konstanter Steifigkeit und Masse muss die Gesamtenergie des Systems konstant bleiben, wenn keine Energie von

außen in das System eingetragen wird. Ist dies der Fall, wird von einem mechanisch konservativen System gesprochen. Sind die Anfangswertbedingungen (vgl. Gleichung (3.16) und (3.17)) bekannt und alle äußeren Kräfte gleich Null, kann die Gesamtenergie aus der Summe der kinetischen und potentiellen Energie ermittelt werden:

$$E_{\text{tot}} = E_{\text{kin}} + E_{\text{pot}} = \frac{1}{2} \dot{\mathbf{u}}^T \mathbf{M} \dot{\mathbf{u}} + \frac{1}{2} \mathbf{u}^T \mathbf{K} \mathbf{u} \quad (3.47)$$

Im Falle einer nicht dissipativen, ungedämpften Simulation muss die Systemenergie über den betrachteten Zeitraum konstant bleiben. Ist dies nicht gewährleistet, ist das numerische Verfahren dissipativ [129]. Nimmt die Energie im Laufe der Berechnung von Zeitschritt zu Zeitschritt ab, wird von numerischer Dämpfung gesprochen. Bei ausreichend langer Simulationszeit wird eine stationäre Lösung erreicht. Beschleunigung und Geschwindigkeit sind dann gleich Null, somit entspricht diese Lösung der statischen Lösung des Systems.

Gelegentlich ist die numerische Dämpfung gewollt und wird eingesetzt, um die Strukturdämpfung zu simulieren oder das numerische Verfahren zu stabilisieren. Nimmt die Energie jedoch zu, schwingt das System auf und verhält sich unphysikalisch. Grund dafür ist zumeist ein zu groß gewählter Zeitschritt Δt oder ein Fehler bei der Bestimmung der Integrationsparameter.

Wird während der Simulation Energie, zum Beispiel durch äußere Kräfte, eingetragen oder entzogen, ist dies bei der Energiebilanz zu berücksichtigen. Das Produkt aus Kraft und Zeitschrittlänge liefert den Impuls für den aktuellen Zeitschritt. Dieser wird mit der aktuellen Geschwindigkeit multipliziert, um die Energie aufgrund externer Beanspruchungen zu bestimmen. Der zusätzlich zu berücksichtigende Energieterm wird dann über die Summe der externen Energien je Zeitschritt ermittelt und von der zuvor bestimmten Energie abgezogen, sodass die Gesamtenergie auch bei Einwirken externer Kräfte konstant bleibt. Es gilt

$$E_{\text{ext}} = \sum_{n=1}^T \mathbf{f}_n \Delta t \dot{\mathbf{u}}_n \approx \sum_{n=1}^T \mathbf{f}_n (\mathbf{u}_n - \mathbf{u}_{n-1}) = \sum_{n=1}^T \mathbf{f}_n \Delta \mathbf{u}_n. \quad (3.48)$$

3.3. Scaled Boundary Finite Element Method

Wie bereits eingangs beschrieben wird die *Scaled Boundary Finite Element Method* zur Simulation des unendlichen Halbraumes im Zeitbereich genutzt. Es wird zunächst die Herleitung der SBFEM im Frequenzbereich beschrieben, anschließend erfolgt die Transformation in den Zeitbereich. Auch hier sei auf die ausführliche Beschreibung und Herleitung von WOLF und SONG verwiesen [162].

Die *scaled boundary* finiten Elemente werden, wie in Abbildung 3.2 dargestellt, mit Hilfe eines lokalen η, ζ Koordinatensystems auf dem Gebietsrand sowie einer radialen Koordinate ξ beschrieben. ξ beschreibt den Abstand zwischen Skalierungszentrum ($\xi = 0$) und dem unendlichen Halbraum ($\xi = \infty$). Auf dem Gebietsrand, der Schnittstelle von Nah- und Fernfeld ist $\xi = 1$. Durch Transformation lassen sich Verschiebungen, Dehnungen und Spannungen aus dem

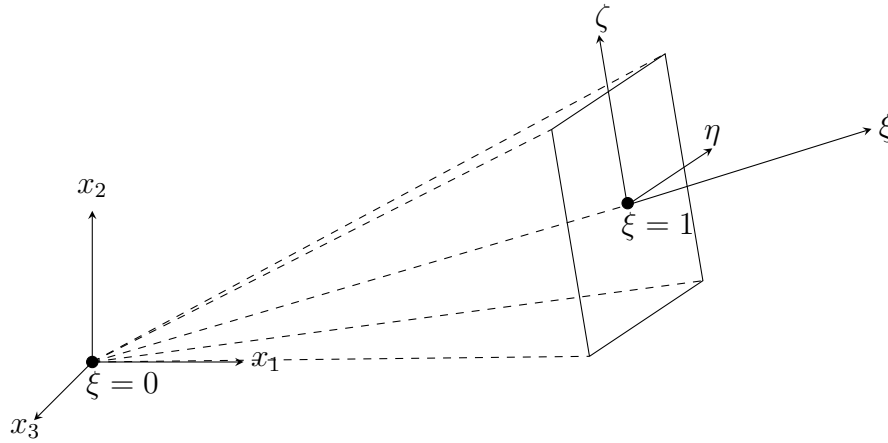


Bild 3.2.: Koordinatentransformation eines dreidimensionalen *scaled boundary* finiten Elements.

lokalen in das globale, kartesische Koordinatensystem übertragen. Die Koordinatentransformation wurde von WOLF und SONG [161, 162] sowie LEHMANN [98] ausführlich beschrieben und soll hier nicht weiter betrachtet werden.

3.3.1. Frequenzbereich

Die Bewegungsdifferentialgleichung (2.7) ist mit dem zeitharmonischen Ansatz (2.31) im Frequenzbereich beschrieben durch:

$$\mathbf{D}^T \hat{\boldsymbol{\sigma}} + \hat{\mathbf{f}} + \omega^2 \rho \hat{\mathbf{u}} = 0. \quad (3.49)$$

Dabei ist \mathbf{D} die Operatormatrix, die Vektoren $\hat{\boldsymbol{\sigma}}$, $\hat{\mathbf{f}}$ und $\hat{\mathbf{u}}$ repräsentieren Spannungstensor, Kräfte und Verschiebungen. Die skalaren Größen ω und ρ beschreiben Kreisfrequenz und Materialdichte. Der in Gleichung (2.7) auftauchende Dämpfungsterm kann bei der Formulierung der SBFEM vernachlässigt werden, da dieser lediglich zur Beschreibung des offenen Randes herangezogen wird und somit die Dämpfungseigenschaften des Fernfeldes keinen Einfluss auf die Lösung haben. Die SBFEM wird hier analog zur vorangegangenen FEM in einer so genannten Kraft-Verschiebungs-Beziehung formuliert. Für jeden Elementknoten, der auf dem Rand Γ_ξ liegt, kann die Verschiebung $\mathbf{u}(\xi)$ mit Hilfe der Ansatzfunktion $N(\eta, \zeta)$ bestimmt werden. Somit gilt

$$\mathbf{u}(\xi, \eta, \zeta) = N(\eta, \zeta) \mathbf{u}(\xi), \quad (3.50)$$

wodurch Dehnungen

$$\epsilon(\xi, \eta, \zeta) = \mathbf{B}_1 \mathbf{u}(\xi)_{,\xi} + \frac{1}{\xi} \mathbf{B}_2 \mathbf{u}(\xi) \quad (3.51)$$

und Spannungen

$$\boldsymbol{\sigma}(\xi, \eta, \zeta) = \mathbf{E} \epsilon(\xi, \eta, \zeta) \quad (3.52)$$

analog zu Gleichung (2.2) und (2.4) definiert sind. \mathbf{B}_1 und \mathbf{B}_2 sind Operatormatrizen, die die Ansatzfunktionen der Elemente sowie deren Ableitungen enthalten. Werden die Gleichungen (3.50), (3.51) und (3.52) in (3.49) eingesetzt und Gleichung (3.49) partiell integriert, folgt nach Einführen der Element-Koeffizientenmatrizen

$$\mathbf{C}_1 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_1^T \mathbf{E} \mathbf{B}_1 |\mathbf{J}| d\eta d\zeta, \quad (3.53)$$

$$\mathbf{C}_2 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_2^T \mathbf{E} \mathbf{B}_1 |\mathbf{J}| d\eta d\zeta, \quad (3.54)$$

$$\mathbf{C}_3 = \int_{-1}^{+1} \int_{-1}^{+1} \mathbf{B}_2^T \mathbf{E} \mathbf{B}_2 |\mathbf{J}| d\eta d\zeta \quad \text{und} \quad (3.55)$$

$$\mathbf{M} = \int_{-1}^{+1} \int_{-1}^{+1} \rho \mathbf{N}^T \mathbf{N} |\mathbf{J}| d\eta d\zeta \quad (3.56)$$

die *scaled boundary* finite Elemente Gleichung in Weggrößenformulierung

$$\mathbf{C}_1 \xi^2 \mathbf{u}_{,\xi\xi}(\xi) + (2\mathbf{C}_1 - \mathbf{C}_2 + \mathbf{C}_2^T) \xi \mathbf{u}_{,\xi}(\xi) + (\mathbf{C}_2^2 - \mathbf{C}_3) \mathbf{u}(\xi) + \omega^2 \mathbf{M} \xi^2 \mathbf{u}(\xi) + \xi^2 \mathbf{f}_b(\xi) = \mathbf{0}. \quad (3.57)$$

Diese ist die schwache Form der Bewegungsdifferentialgleichung und gültig für das Intervall $0 \leq \xi \leq \infty$. Für die Kopplung von SBFEM und FEM wird die effektive Steifigkeitsmatrix \mathbf{S}^∞ auf dem Gebietsrand ($\xi = 1$) eingeführt und Gleichung (3.57) in

$$(\mathbf{S}^\infty(\omega) + \mathbf{C}_2) \mathbf{C}_1^{-1} (\mathbf{S}^\infty(\omega) + \mathbf{C}_2^T) - \mathbf{S}^\infty(\omega) - \omega \mathbf{S}^\infty(\omega)_{,\omega} - \mathbf{C}_3 + \omega^2 \mathbf{M} = \mathbf{0} \quad (3.58)$$

überführt. Diese nichtlineare Differentialgleichung erster Ordnung beschreibt die *scaled boundary* finite Elemente Gleichung im Frequenzbereich für die räumliche Elastodynamik. Die frequenzabhängige effektive Steifigkeitsmatrix des unendlichen Halbraums ist gegeben durch $\mathbf{S}^\infty(\omega)$ [162].

3.3.2. Zeitbereich

Für Untersuchungen im Zeitbereich wird die *acceleration unit impulse response matrix* oder Einflussmatrix $\mathbf{M}^\infty(t)$ benötigt, um den Einfluss des Fernfeldes auf das Nahfeld zu bestimmen, dafür wird in Gleichung (3.58)

$$\mathbf{M}^\infty(\omega) = \frac{\mathbf{S}^\infty(\omega)}{(i\omega)^2} \quad (3.59)$$

eingesetzt und eine Fourier-Transformation aus dem Frequenzbereich in den Zeitbereich durchgeführt. Das Ergebnis der Transformation ist

$$\int_0^t \tilde{\mathbf{M}}^\infty(t-\tau) \tilde{\mathbf{M}}^\infty(\tau) d\tau + \tilde{\mathbf{C}}_2 \int_0^t \int_0^\tau \tilde{\mathbf{M}}^\infty(\tau') d\tau' d\tau + \int_0^t \int_0^\tau \tilde{\mathbf{M}}^\infty(\tau') d\tau' d\tau \tilde{\mathbf{C}}_2^T + t \int_0^t \tilde{\mathbf{M}}^\infty(\tau) d\tau - \frac{t^3}{6} \tilde{\mathbf{C}}_3 \mathbf{H}(t) - t \tilde{\mathbf{M}} \mathbf{H}(t) = \mathbf{0}, \quad (3.60)$$

mit den Koeffizientenmatrizen

$$\mathbf{C}_1 = \mathbf{L} \mathbf{L}^T, \quad (3.61)$$

$$\tilde{\mathbf{C}}_2 = \mathbf{L}^{-1} \mathbf{C}_2 \mathbf{L}^{-T}, \quad (3.62)$$

$$\tilde{\mathbf{C}}_3 = \mathbf{L}^{-1} (\mathbf{C}_3 - \mathbf{C}_2 \mathbf{C}_1^{-1} \mathbf{C}_2^T) \mathbf{L}^{-T}, \quad (3.63)$$

$$\tilde{\mathbf{M}} = \mathbf{L}^{-1} \mathbf{M} \mathbf{L}^{-T} \quad (3.64)$$

und der Heaviside-Funktion $\mathbf{H}(t)$. Die Matrix \mathbf{L} ist die untere Dreiecksmatrix einer Cholesky-Zerlegung der Matrix \mathbf{C}_1 , siehe Gleichung (3.61). In anderen Quellen (wie beispielsweise [32, 98]) ist diese Substitution anhand der oberen Dreiecksmatrix beschrieben. Da die hier implementierte Software die untere Dreiecksmatrix verwendet, wurde dies hier dementsprechend angepasst. Die zeitabhängige Matrix $\tilde{\mathbf{M}}^\infty(t)$ wird für jeden Zeitschritt mit der Gleichung (3.60) bestimmt. Die gesuchte Einflussmatrix $\mathbf{M}^\infty(t)$ wird letztlich mit

$$\mathbf{M}^\infty(t) = \mathbf{L} \tilde{\mathbf{M}}^\infty(t) \mathbf{L}^T \quad (3.65)$$

berechnet und muss für jeden betrachteten Zeitschritt bestimmt werden. Da die Berechnungsvorschrift zur Bestimmung der Einflussmatrix im ersten Zeitschritt von allen darauf folgenden Zeitschritten abweicht, wird der Ansatz zur Bestimmung der Matrix für den ersten bzw. n -ten Zeitschritt nachfolgend diskutiert.

3.3.2.1. Erster Zeitschritt

Das Lösen des Faltungsintegrals aus Gleichung (3.60) führt im ersten Zeitschritt zu einer quadratischen Gleichung mit der unbekannten Matrix $\tilde{\mathbf{M}}^\infty$:

$$\tilde{\mathbf{M}}_0^{\infty 2} + \frac{\Delta t}{2} (\tilde{\mathbf{C}}_2 + \mathbf{I}) \tilde{\mathbf{M}}_0^\infty + \tilde{\mathbf{M}}_0^\infty \frac{\Delta t}{2} (\tilde{\mathbf{C}}_2^T + \mathbf{I}) - \frac{\Delta t^2}{6} \tilde{\mathbf{C}}_3 - \tilde{\mathbf{M}} = \mathbf{0} \quad (3.66)$$

Diese Gleichung ist analog zur algebraischen Form der Riccati-Gleichung

$$\mathbf{F} \mathbf{X} + \mathbf{X} \mathbf{F}^T - \mathbf{X} \mathbf{B} \mathbf{B}^T \mathbf{X} + \mathbf{Q} = \mathbf{0}, \quad (3.67)$$

wobei $\mathbf{F} = -\frac{\Delta t}{2} (\tilde{\mathbf{C}}_2 + \mathbf{I})$ und $\mathbf{Q} = \frac{\Delta t^2}{6} \tilde{\mathbf{C}}_3 + \tilde{\mathbf{M}}$ ist. Die Matrix \mathbf{B} entspricht der Einheitsmatrix \mathbf{I} und \mathbf{X} entspricht der gesuchten Lösung $\tilde{\mathbf{M}}_0^\infty$. Die quadratische Form tritt nur im ersten

Zeitschritt auf, wodurch sich die Art der Berechnung von allen nachfolgenden signifikant unterscheidet. Ausführliche Informationen bezüglich der Riccati-Gleichungen und deren Lösungsverhalten sind beispielsweise [132] zu entnehmen. Dort sind ebenfalls eine Reihe an Beweisen und Theoremen zu diesem Gleichungstyp zusammengefasst und erläutert.

3.3.2.2. n -ter Zeitschritt

Alle nachfolgenden Zeitschritte t_n mit $n \geq 1$ sind durch die Gleichung

$$\begin{aligned} \left(\tilde{\mathbf{M}}_0^\infty + \frac{\Delta t}{2} \tilde{\mathbf{C}}_2 \right) \tilde{\mathbf{M}}_n^\infty + \tilde{\mathbf{M}}_n^\infty \left(\tilde{\mathbf{M}}_0^\infty + \frac{\Delta t}{2} \tilde{\mathbf{C}}_2^T \right) + t \tilde{\mathbf{M}}_n^\infty = \\ - \sum_{k=1}^{n-1} \left(\tilde{\mathbf{M}}_{n-k}^\infty \tilde{\mathbf{M}}_k^\infty \right) - \tilde{\mathbf{C}}_2 \left(\frac{\mathbf{J}_{n-1}}{\Delta t} + \mathbf{I}_{n-1} \right) \\ - \left(\frac{\mathbf{J}_{n-1}}{\Delta t} + \mathbf{I}_{n-1} \right) \tilde{\mathbf{C}}_2^T + \frac{t^3}{6\Delta t} \tilde{\mathbf{C}}_3 + \frac{t}{\Delta t} \left(\tilde{\mathbf{M}} - \mathbf{I}_{n-1} \right) \end{aligned} \quad (3.68)$$

unter der Berücksichtigung der Integrale

$$\mathbf{I}_n = \int_0^{n\Delta t} \tilde{\mathbf{M}}^\infty(\tau) d\tau = \mathbf{I}_{n-1} + \Delta t \tilde{\mathbf{M}}_n^\infty \quad (3.69)$$

und

$$\mathbf{J}_n = \int_0^{n\Delta t} \int_0^\tau \tilde{\mathbf{M}}^\infty(\tau') d\tau' d\tau = \mathbf{J}_{n-1} + \Delta t \mathbf{I}_{n-1} + \frac{\Delta t^2}{2} \tilde{\mathbf{M}}_n^\infty \quad (3.70)$$

beschrieben. Gleichung (3.68) lässt sich in eine Lyapunov-Gleichung der Form

$$\tilde{\mathbf{A}}\mathbf{X} + \mathbf{X}\tilde{\mathbf{A}}^T = \mathbf{C} \quad (3.71)$$

überführen, wenn $\tilde{\mathbf{A}} = \tilde{\mathbf{M}}_0^\infty + \frac{\Delta t}{2} \tilde{\mathbf{C}}_2 + \frac{t}{2} \mathbf{I}$ eingeführt und in die Gleichung eingesetzt wird. Matrix \mathbf{C} repräsentiert die rechte Seite der Gleichung (3.68). \mathbf{X} entspricht auch hier der gesuchten Lösung $\tilde{\mathbf{M}}_n^\infty$. Nachdem alle $\tilde{\mathbf{M}}_n^\infty$ -Matrizen bestimmt sind, können diese mit Gleichung (3.65) in die für die Kopplung benötigten Matrizen \mathbf{M}_n^∞ überführt werden.

3.4. Kopplung von FEM und SBFEM

Nachdem in den vorangegangenen Abschnitten die Grundlagen der FEM sowie der SBFEM dargestellt wurden, soll nun die Kopplung von FEM und SBFEM näher erläutert werden. Es handelt sich um eine bidirektionale, auch als „stark“ bezeichnete Kopplung der beiden Teilsysteme. Zur besseren Veranschaulichung werden hier die Einträge innerhalb der Matrizen und Vektoren in Nah- und Fernfeld sortiert:

$$\begin{bmatrix} \mathbf{M}_{\Omega\Omega} & \mathbf{M}_{\Omega\Gamma} \\ \mathbf{M}_{\Gamma\Omega} & \mathbf{M}_{\Gamma\Gamma} \end{bmatrix} \ddot{\mathbf{u}} + \begin{bmatrix} \mathbf{C}_{\Omega\Omega} & \mathbf{C}_{\Omega\Gamma} \\ \mathbf{C}_{\Gamma\Omega} & \mathbf{C}_{\Gamma\Gamma} \end{bmatrix} \dot{\mathbf{u}} + \begin{bmatrix} \mathbf{K}_{\Omega\Omega} & \mathbf{K}_{\Omega\Gamma} \\ \mathbf{K}_{\Gamma\Omega} & \mathbf{K}_{\Gamma\Gamma} \end{bmatrix} \mathbf{u} = \begin{bmatrix} \mathbf{p}_{\Omega\Omega} \\ \mathbf{p}_{\Gamma\Gamma} \end{bmatrix} - \begin{bmatrix} \mathbf{0} \\ \mathbf{p}_b \end{bmatrix}. \quad (3.72)$$

Bei einer so gewählten Aufteilung entstehen nun innerhalb der Matrizen und Vektoren Blöcke, die sich eindeutig dem Nahfeld $\Omega\Omega$ oder dem Fernfeld $\Gamma\Gamma$ zuordnen lassen. Die Blöcke, die mit $\Omega\Gamma$ und $\Gamma\Omega$ gekennzeichnet sind, koppeln die Freiheitsgrade von Nah- und Fernfeld. Der hier neu eingeführte Vektor \mathbf{p}_b greift ausschließlich am Rand Γ zwischen Nah- und Fernfeld an (vgl. Bild 1.2) und beschreibt den Einfluss des unendlichen Halbraumes auf das Nahfeld, wobei dieser Einfluss als angreifende Kraft auf das finite Elemente Netz wirkt.

Die erste Substruktur, das Nahfeld, wird mit Hilfe der FEM beschrieben. Die zweite Substruktur, mit der das Fernfeld diskretisiert ist, wird mit Hilfe der SBFEM gelöst. Die am Interface zwischen Nah- und Fernfeld wirkenden Kräfte werden durch das Faltungsintegral

$$\mathbf{p}_b(t) = \int_0^t \mathbf{M}^\infty(t - \tau) \ddot{\mathbf{u}}(\tau) d\tau \quad (3.73)$$

beschrieben. Hierbei ist $\mathbf{M}^\infty(t)$ die Einheits-Beschleunigungsmatrix, die in der Literatur auch Einflussmatrix genannt wird. Zum Lösen des Faltungsintegrals (3.73) wird davon ausgegangen, dass die Einflussmatrizen \mathbf{M}_i^∞ innerhalb eines Zeitschritts der Länge Δt konstant sind,

$$\mathbf{M}^\infty(t) = \begin{cases} \mathbf{M}_0^\infty & t \in [0; \Delta t], \\ \mathbf{M}_1^\infty & t \in [\Delta t; 2\Delta t], \\ \vdots & \vdots \\ \mathbf{M}_{n-j}^\infty & t \in [(n-j-1)\Delta t; (n-j)\Delta t], \\ \vdots & \vdots \\ \mathbf{M}_{n-1}^\infty & t \in [(n-2)\Delta t; (n-1)\Delta t], \\ \mathbf{M}_n^\infty & t \in [(n-1)\Delta t; n\Delta t]. \end{cases} \quad (3.74)$$

Aufgrund dieser Annahme lässt sich Gleichung (3.73) folgendermaßen umformen

$$\mathbf{p}_b(t_n) = \sum_{j=1}^n \mathbf{M}_{n-j}^\infty \int_{(j-1)\Delta t}^{j\Delta t} \ddot{\mathbf{u}}(\tau) d\tau \quad (3.75)$$

und unter Anwendung des Zeitintegrationsverfahrens in

$$\mathbf{p}_b(t_n) = \gamma \Delta t \mathbf{M}_0^\infty \ddot{\mathbf{u}}_n + \sum_{j=1}^{n-1} \mathbf{M}_{n-j}^\infty (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}) \quad (3.76)$$

überführen. Somit wird, anstatt das Integral zu lösen, lediglich die Differenz der Geschwindigkeiten gebildet und mit der zugehörigen Einflussmatrix multipliziert und summiert. Die Kopplung von FEM und SBFEM erfolgt dann an der gemeinsamen Schnittstelle zwischen Nah- und Fernfeld. Die Anteile aus der SBFEM können in dieser Formulierung einfach zu den Einträgen

der FEM in der Gleichung (3.72) addiert werden

$$\begin{aligned} & \begin{bmatrix} \mathbf{M}_{\Omega\Omega} & \mathbf{M}_{\Omega\Gamma} \\ \mathbf{M}_{\Gamma\Omega} & \mathbf{M}_{\Gamma\Gamma} + \gamma\Delta t\mathbf{M}_0^\infty \end{bmatrix} \ddot{\mathbf{u}} + \begin{bmatrix} \mathbf{C}_{\Omega\Omega} & \mathbf{C}_{\Omega\Gamma} \\ \mathbf{C}_{\Gamma\Omega} & \mathbf{C}_{\Gamma\Gamma} \end{bmatrix} \dot{\mathbf{u}} + \\ & \begin{bmatrix} \mathbf{K}_{\Omega\Omega} & \mathbf{K}_{\Omega\Gamma} \\ \mathbf{K}_{\Gamma\Omega} & \mathbf{K}_{\Gamma\Gamma} \end{bmatrix} \mathbf{u} = \begin{bmatrix} \mathbf{p}_{\Omega\Omega} \\ \mathbf{p}_{\Gamma\Gamma} - \sum_{j=1}^{n-1} \mathbf{M}_{n-j}^\infty (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}) \end{bmatrix}, \end{aligned} \quad (3.77)$$

wodurch die mathematische Darstellung der FEM-SBFEM Kopplung vollständig beschrieben ist.

Wird, wie in Abschnitt 6.1 beschrieben, davon ausgegangen, dass sich ein lineares Verhalten für \mathbf{M}^∞ zum Zeitpunkt t_m einstellt, sei darauf hingewiesen, dass die Berechnung von $\mathbf{p}_b(t_n)$ entsprechend anzupassen ist (vgl. Gleichungen (6.3) bis (6.8)).

4. Wissenschaftliches Rechnen

Der Fokus dieser Arbeit liegt in der Erstellung einer effizienten, gekoppelten Software zur Simulation randloser Gebiete auf Basis eines gekoppelten Ansatzes von Finiter Elemente Methode und *Scaled Boundary Finite Element Method* mit dem Ziel, realitätsnahe bzw. praxistaugliche Analysen durchführen zu können. Da dies oftmals mit einer großen Anzahl an Freiheitsgraden einhergeht und mit der Größe des zu diskretisierenden Gebietes und dessen Detailgrades in der Regel der Berechnungsaufwand und damit die Rechenzeit zunimmt, gilt es hier Möglichkeiten zu finden, die geeignet sind, diese rechenzeitintensiven Simulationen dennoch in einer angemessenen Zeit durchführen zu können.

Probleme dieser Art werden im Allgemeinen unter dem Begriff des wissenschaftlichen Rechnens zusammengefasst. Dabei ist wissenschaftliches Rechnen als Schnittmenge aus numerischer Mathematik, Computerwissenschaften und Modellierung zu verstehen [85]. Das wissenschaftliche Rechnen erweitert das klassische Modell der Forschung in den Natur- und Ingenieurwissenschaften von zwei auf drei Säulen. Die klassische Forschung basiert auf der Theorie und dem Experiment. Als dritte Säule kommt durch das wissenschaftliche Rechnen die numerische Simulation hinzu. Ziel ist es dabei, zum einen die Anzahl der möglicherweise sehr teuren Experimente zu verringern oder vollständig zu ersetzen und zum anderen Experimente durchzuführen, die in der Realität technisch nicht möglich oder schlicht unwirtschaftlich sind.

Je nach Fragestellung und Anwendungsgebiet sind wissenschaftliche numerische Experimente sehr rechenintensiv und produzieren große Mengen an auszuwertenden Daten. Diese können wenige Megabyte bis hin zu vielen hundert Gigabyte umfassen. Um die erforderlichen Rechenressourcen zur Verfügung zu stellen, finden wissenschaftliche Rechnungen in der Regel auf so genannten Supercomputern oder auch verteilten Systemen statt. Als Supercomputer wird zumeist die leistungsstärkste Maschine, die zurzeit zur Verfügung steht, bezeichnet, weshalb heutige Supercomputer oftmals die Desktop und Notebook Computer von morgen sind. Verteilte Systeme hingegen setzen sich im Allgemeinen aus einer großen Anzahl von Standardkomponenten, die durch eine sehr leistungsstarke Netzwerktechnik verbunden sind, zu so genannten Rechenclustern zusammen und stellen somit mit standardisierter Hardware sehr viel Rechenkapazität zur Verfügung.

Im Vergleich zu Supercomputern bieten verteilte Systeme den Vorteil, dass die Anschaffungskosten aufgrund der Verwendung von Standardkomponenten geringer ausfallen. Hinzu kommt, dass diese Systeme in nahezu beliebiger Größe aufgebaut werden können und es möglich ist, diese verteilten Systeme bei Bedarf sogar zu erweitern, wodurch verteilte Systeme flexibel eingesetzt werden können und heutzutage weit verbreitet sind. Deshalb wird im Folgenden näher auf die Funktionsweise des verteilten oder auch parallelen Rechnens eingegangen.

4.1. Paralleles und verteiltes Rechnen

Verteiltes bzw. paralleles Rechnen wird heute häufig mit den Begriffen Hochleistungsrechnen, *High Performance Computing* (HPC) oder auch *Supercomputing* in Verbindung gebracht [50, 85]. Dabei geht es darum, ein zu lösendes Problem entsprechend aufzuteilen und verteilt auf mehrere Prozesse gleichzeitig zu lösen. Begründet ist dieser Ansatz in der limitierten Leistungsfähigkeit heutiger Computer und den immer komplexer werdenden Simulationen. Das parallele Lösen eines gegebenen Problems ermöglicht die Berechnung einer Lösung auch dann durchzuführen, wenn dieses nicht auf einem einzigen System geschehen kann, da z. B. nicht genügend Arbeitsspeicher zur Verfügung steht oder die Rechenzeit das zulässige Zeitfenster überschreitet. Zum anderen bietet das verteilte Rechnen die Möglichkeit, Fragestellungen mit einem höheren Detailgrad und somit mehr Freiheitsgraden zu untersuchen, ein und dasselbe System in kürzerer Zeit zu berechnen oder mehrere Testfälle zeitgleich abzuarbeiten, da mehr Rechenleistung bereitgestellt wird.

Bereits 1965 sagte MOORE auf Grundlage empirischer Beobachtung voraus, dass sich die Anzahl der Transistoren in integrierten Schaltungen und somit die Rechenleistung jährlich verdoppeln werden würde [116]. Diese Aussage wurde in den Folgejahren leicht korrigiert, bereits 1975 zeigte sich jedoch, dass eine Verdoppelung der Rechenleistung innerhalb eines Zeitraums von zwei Jahren als durchaus realistisch anzusehen war [137]. Diese Abschätzung wurde als *Moore's law*, dessen Gesetzmäßigkeit bis heute gültig ist, bekannt. Mit der Anzahl der Transistoren steigt gleichzeitig die Anzahl der möglichen Rechenoperationen und somit die Leistung der Prozessoren. Dennoch reicht diese Leistungssteigerung der Prozessoren für heutige rechenintensive Simulationen nicht aus, weshalb auf verteilte Systeme mit vielen Prozessoren zurückgegriffen werden muss, wenn große Problemstellungen zu lösen sind.

Der Hauptprozessor, die datenverarbeitende Logik-Einheit (auch *central processing unit* CPU genannt), bildet das zentrale Rechenwerk in einem Computer und entscheidet zusammen mit der Bandbreite des Systems maßgeblich über die Rechengeschwindigkeit eines Computers. Die Bandbreite beschreibt den Datendurchsatz von Speicher und Netzwerk, je größer die Bandbreite ist desto leistungsfähiger ist das gesamte System. Neben der Bandbreite ist auch die Latenz ein wichtiger Faktor. Die Latenz ist die Ansprechzeit des Speichers sowie des Netzwerks und beschreibt die Zeit, die verstreicht, bis angeforderte Daten zur Bearbeitung zur Verfügung stehen. Bandbreite und Latenz hängen von Hardware und Software (z. B. Netzwerk-Protokoll) ab.

Die maximale Leistung eines Systems ist die *Peakperformance* (CPU-Leistung), die in FLOPS (*FL*oatingpoint *O*perations *P*er *S*econd) gemessen wird. Die *Peakperformance* beschreibt die maximal durchführbare Anzahl an Rechenoperationen. Bei realen Anwendungen wird die theoretisch mögliche *Peakperformance* in der Regel nicht erreicht. Grund dafür ist, dass die effektiv nutzbare Speicherbandbreite in der Regel nicht ausreicht die CPU mit genügend Daten zu versorgen, damit in jedem Takt gerechnet wird. Die Speicher- und Netzwerkzugriffszeiten betragen ein Vielfaches der CPU Taktzyklen und bremsen das System somit aus. Die *Peakperformance*

sowie die reale Performance aktueller Computersysteme wird beispielsweise in der so genannten TOP500 zusammengestellt und kann unter www.top500.org abgefragt werden.

Bei parallelen und verteilten Systemen wird unterschieden in *shared memory* und *distributed memory* Systeme. Bei *shared memory* Systemen teilen sich alle Prozessoren einen gemeinsamen Speicher, auf den sie alle gemeinsam zugreifen können. Dies ermöglicht eine recht einfache Entwicklung der Algorithmen, da sämtliche Informationen jederzeit für alle Prozesse zu Verfügung stehen. Eine Steuerung der Datenzugriffe ist allerdings unerlässlich, damit nicht gleichzeitig von mehreren Prozessen in denselben Speicherbereich geschrieben wird und somit ein Datenverlust droht. Anwendungen, die für *shared memory* Systeme entwickelt werden, laufen ausschließlich auf derartigen Systemen. Bei *distributed memory* Systemen hingegen hat jeder Prozess seinen eigenen Arbeitsspeicher oder Arbeitsspeicherbereich, auf den ausschließlich der zugeordnete Prozess zugreifen kann. Eine Koordination der Speicherzugriffe ist somit nicht nötig, stattdessen müssen die Prozesse nun untereinander kommunizieren, um benötigte Informationen auszutauschen. Programme, die für *distributed memory* Systeme entwickelt werden, laufen ebenfalls auf *shared memory* Systemen und sind somit vielseitiger einsetzbar.

Möglichkeiten der Umsetzung paralleler Software gibt es reichlich. Der einfachste Weg ist die Nutzung selbst-parallelisierender Compiler wie beispielsweise *High Performance Fortran* [88, 110]. Eine Alternative ist der Einsatz einer anweisungsbasierten Parallelisierung, wie sie mit Hilfe von OpenMP [121] oder gegebenenfalls sogar auf Compiler- oder Systemebene realisiert werden kann. Beide Verfahren funktionieren allerdings lediglich auf *shared memory* Systemen. Auf *distributed memory* Systemen lässt sich die Kommunikation zwischen den Prozessen beispielsweise mit *Parallel Virtual Maschine* (PVM) [63] oder auch dem *Message Passing Interface* (MPI) [117] realisieren. Die Kommunikation zwischen den Prozessen, um den Datenaustausch zu ermöglichen, muss vom Programmentwickler, abhängig von dem gewählten Verfahren, zusätzlich zur eigentlichen Problemstellung berücksichtigt und implementiert werden.

Fällt die Entscheidung, wie in diesem Fall die Implementierung für verteilte Systeme (Computercluster) durchzuführen, dann ist das entsprechende Konzept der Parallelisierung zu wählen. Die Umsetzung erfolgt nach dem so genannten *Multiple Instruction Multiple Data* (MIMD) oder *Multiple Programm Multiple Data* (MPMD) Konzept. Dabei arbeiten mehrere Programme mit den jeweils zugeteilten Teildatensätzen auf jeweils einem Prozess. Diese Teildatensätze sind in der Regel innerhalb jedes Prozesses einzigartig. Erfordert das Abarbeiten eines Teildatensatzes, z. B. Matrix-Vektor- oder Matrix-Matrix-Multiplikation oder auch das Lösen eines Gleichungssystems, Informationen eines anderen Teildatensatzes, der nicht lokal (auf dem laufenden Prozess) zur Verfügung steht, ist eine Kommunikation zwischen den Prozessen nötig, um das Abarbeiten der Teildatensätze zu ermöglichen.

Bei der Kommunikation zwischen den einzelnen Prozessen ist es notwendig, diese zeitlich aufeinander abzustimmen, d. h. zu synchronisieren [92]. Damit eine Kommunikation zwischen den Prozessen erfolgreich durchgeführt werden kann, werden ein sender Prozess, der eine Information verteilt und mindestens ein empfangender Prozess, der die Information empfängt,

benötigt. Für eine fehlerfreie Kommunikation von Datenpaketen muss sichergestellt werden, dass sendende und empfangende Prozesse genau die Daten austauschen, die sie von einander erwarten, da es sonst zu Datenverlusten und somit zu Fehlern in der Kommunikation und ggf. auch im weiteren Programmablauf kommt. Eine effiziente und umfangreiche Umsetzung zur Kommunikation zwischen Prozessen bietet die zuvor genannte Bibliothek MPI. Wie bereits erwähnt, ist die Kommunikation zwischen den Prozessen unerlässlich, um gemeinsam eine Problemstellung zu lösen. Während dieser Kommunikation wird allerdings nicht das eigentliche Problem gelöst, sondern lediglich dafür gesorgt, dass im nächsten Programmschritt damit fortgefahren werden kann. Es kommt durch die Kommunikation somit zu dem so genannten *overhead* im Programmablauf. Die für die Synchronisation und Kommunikation benötigte Rechenzeit geht dem eigentlichen Lösungsprozess verloren. Ziel muss es also immer sein, den Anteil der Synchronisation und Kommunikation so weit wie möglich zu reduzieren, um eine möglichst effiziente Software zu entwickeln. Dies wird in Kapitel 7 genauer veranschaulicht.

4.2. Softwareimplementierung

Für die Simulation der Boden-Bauwerk-Interaktion werden zwei Softwareprogramme entwickelt, bzw. weiterentwickelt und in Kapitel 8 am Beispiel diskutiert. Das Fernfeld wird mit dem Programm SCABO, ein auf der *Scaled Boundary* Finite Elemente Methode basierendes Programm zur Modellierung und Simulation randloser Gebiete, berechnet [142]. Für das gekoppelte System aus Nah- und Fernfeld wird das Softwarepaket ELPASO, ein auf der Finiten Elemente Methode basierendes Programm, verwendet [17]. Beide Softwareprogramme werden in den nachfolgenden Abschnitten ausführlich diskutiert.

Um den Entwicklungsaufwand so gering wie möglich zu gestalten, sollten bereits existierende und umfassend getestete Bibliotheken eingebunden werden. Dabei gilt in der Regel, je weiter eine Bibliothek verbreitet ist, desto zuverlässiger und performanter ist sie. Es kann davon ausgegangen werden, dass insbesondere weit verbreitete Bibliotheken nahezu fehlerfrei arbeiten und laufend optimiert und aktualisiert werden. Durch die stetige Optimierung der Bibliotheken steigt oftmals auch die Leistungsfähigkeit der eigenen Software.

Die entwickelten Softwarepakete greifen aufgrund der zuvor aufgeführten Gründe auf externe Bibliotheken zurück. Zur besseren Übersichtlichkeit werden daher in den nachfolgenden Abschnitten die in den hier verwendeten Softwareprogrammen eingebundenen externen Bibliotheken kurz vorgestellt. Vertiefende Informationen zu den externen Bibliotheken sind in den jeweiligen Entwicklerhandbüchern zu finden.

4.2.1. Scaled Boundary SCABO

SCABO ist ein entwickeltes *Scaled Boundary* Finite Elemente Programm [142], mit dem die Einflussmatrizen eines unendlichen Halbraums für die Boden-Bauwerk-Interaktion berechnet werden. Es handelt sich bei diesem Programm um eine klassische Eingabe-Verarbeitung-Ausgabe (EVA) Anwendung, die mittels einer Eingabedatei sowie zusätzlicher Steuerungsbefehle über die Konsole vom Anwender aufgerufen wird. Ein Eingreifen des Anwenders auf einen laufenden Prozess ist somit während der Laufzeit nicht möglich. Alle in dieser Arbeit diskutierten Berechnungen können ab der Version 315 durchgeführt werden [142].

In der Eingabedatei sind das zu berechnende Netz samt Knoten und Elementen, den Elementen zugewiesene Materialien sowie das Skalierungszentrum definiert. Hinzu kommen Informationen für den Löser, wie beispielsweise Zeitschrittlänge und Anzahl der Zeitschritte. Über die optionalen Steuerungsbefehle lassen sich spezielle Lösungsverfahren, das Format der Ausgabedateien oder auch die Berechnung und Ausgabe von Residuen steuern.

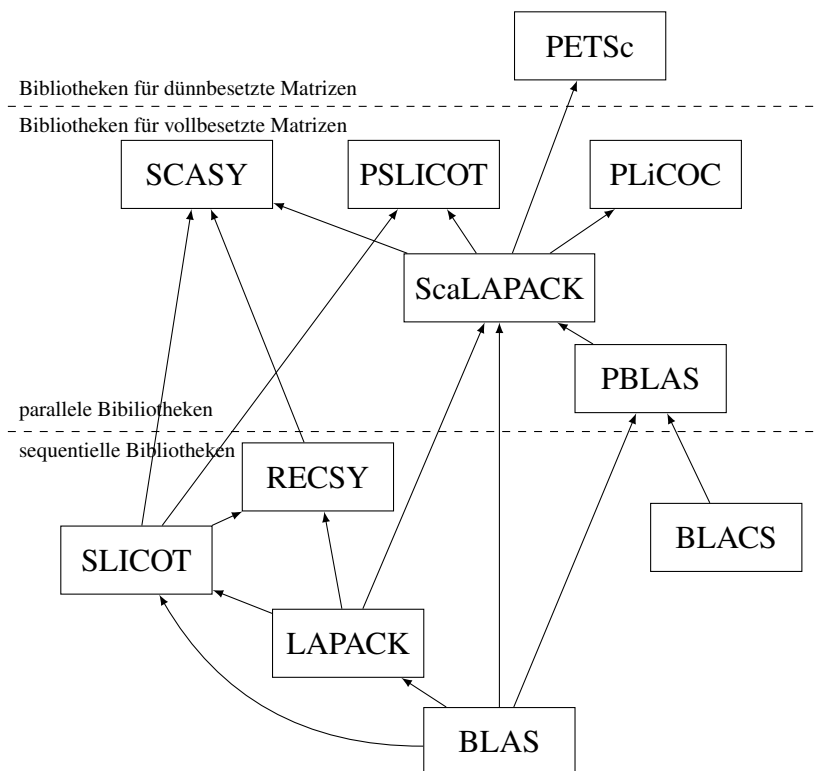


Bild 4.1.: In SCABO eingebundene externe Bibliotheken.

Die Software greift auf eine Vielzahl externer Bibliotheken zurück, wodurch so auf bereits validierte Algorithmen zurückgegriffen werden konnte und somit die Entwicklungszeit verkürzt wurde. Die eingesetzten externen Bibliotheken sind samt Abhängigkeiten in Bild 4.1 dargestellt. Je nach Einsatzgebiet wurden diese Bibliotheken für sequentielle oder parallele Softwareentwicklung sowie vollbesetzte bzw. dünnbesetzte Gleichungssysteme entwickelt. In dem Bild grenzt die obere gestrichelte Linie Bibliotheken, die speziell für vollbesetzte beziehungs-

weise dünnbesetzte Matrizen entwickelt wurden, voneinander ab. Die untere gestrichelte Linie trennt die sequentiellen von den parallel arbeitenden Bibliotheken.

Die Berechnung des Fernfeldes im Zeitbereich erfolgt, wie in Abschnitt 3.3 beschrieben, im Wesentlichen mit Hilfe vollbesetzter Matrizen. Aus diesem Grund bildet die Bibliothek *Scalable Linear Algebra PACKage* (ScaLAPACK) [29, 136], die speziell für die Verarbeitung vollbesetzter Matrizen auf verteilten Systemen entwickelt wurde, den Kern dieser Software. Die Manipulation der dünnbesetzten Matrizen erfolgt mit dem *Portable, Extensible Toolkit for Scientific Computation* (PETSc) [7–9]. PETSc und ScaLAPACK kommunizieren über eine implementierte Schnittstelle, die bereits in [143] ausführlich beschrieben wurde. Es wurden so genannte *Wrapper*-Klassen entwickelt, die die Übertragung der Datenstrukturen in die unterschiedlichen Datenformate der einzelnen Bibliotheken übernehmen. Da SCABO in C++ geschrieben wurde, PETSc in C und ScaLAPACK sowie die meisten der anderen Bibliotheken hingegen in *Fortran* entwickelt wurden, war besondere Vorsicht bei der Umsetzung der *Wrapper*-Klassen geboten. Die Kommunikation der parallel arbeitenden Bibliotheken erfolgt mit Hilfe des *Message Passing Interface* (MPI) [117]. Die lediglich im ersten Zeitschritt auftauchende Riccati Gleichung (vgl. Gleichung (3.67), Kapitel 3.3.2.1) wird mit dem in der Bibliothek PLiCOC [25] implementierten iterativen Löser bestimmt. Im Anschluss daran wird mittels PSLICOT die Schur-Zerlegung bestimmt. Für jeden weiteren Zeitschritt erfolgt das Lösen der Lyapunov Gleichung (vgl. Gleichung (3.71), Kapitel 3.3.2.2) mit SCASY [68–70]. Bei dem Lösungsverfahren der Lyapunov Gleichung handelt es sich ebenfalls um einen iterativen Algorithmus. Zusätzlich zu dem allgemeinen Lösungsansatz für Lyapunov Gleichungen bietet SCASY spezielle Algorithmen, die für symmetrische Matrizen entwickelt wurden. Diese Algorithmen können in der aktuellen Version von SCABO ebenfalls zum Berechnen der Einflussmatrizen ausgewählt werden.

Neben den parallelen Bibliotheken kommen auch sequentielle Bibliotheken zum Einsatz, die grundlegende Funktionen der linearen Algebra bereitstellen. Diese sequentiellen Programmteile laufen lokal auf einzelnen Prozessen und werden von den übergeordneten parallelen Algorithmen gesteuert. BLAS (*Basic Linear Algebra Subprograms*) [31] ist in drei Level unterteilt und liefert Algorithmen für Vektor- und Matrixoperationen. Die in Level 1 implementierten Funktionen führen Skalar-Vektor und Vektor-Vektor Operationen aus. Level 2 stellt Matrix-Vektor und Level 3 Matrix-Matrix Operationen zur Verfügung. Aufgrund der effizienten und einfachen Umsetzung bildet BLAS heutzutage die Grundlage vieler Anwendungen. LAPACK (*Linear Algebra PACKage*) [3] basiert auf BLAS und ist eine Erweiterung von Level 3 BLAS. Das Paket ermöglicht den Zugriff auf eine Reihe von Algorithmen zur Matrixzerlegung und zum Lösen von Gleichungssystemen [93]. Diese Bibliotheken werden lokal innerhalb eines Prozesses ausgeführt. Die nötige Kommunikation übernimmt in diesem Fall eine der Bibliotheken mit einem höheren Abstraktionslevel, wie beispielsweise PBLAS (*Parallel Basic Linear Algebra Subprograms*) oder ScaLAPACK (*Scalable Linear Algebra PACKage*). Beide Bibliotheken nutzen die in *Basic Linear Algebra Communication Subprograms* (BLACS) [30, 49] auf MPI basierende standardisierte Schnittstellen zur Kommunikation. Hinzu kommen die Bibliotheken *Subroutine*

Library in Systems and Control Theory (SLICOT) [24, 146] und (RECSY) [82–84], die grundlegende Funktionen für SCASY und PSLICOT bereitstellen.

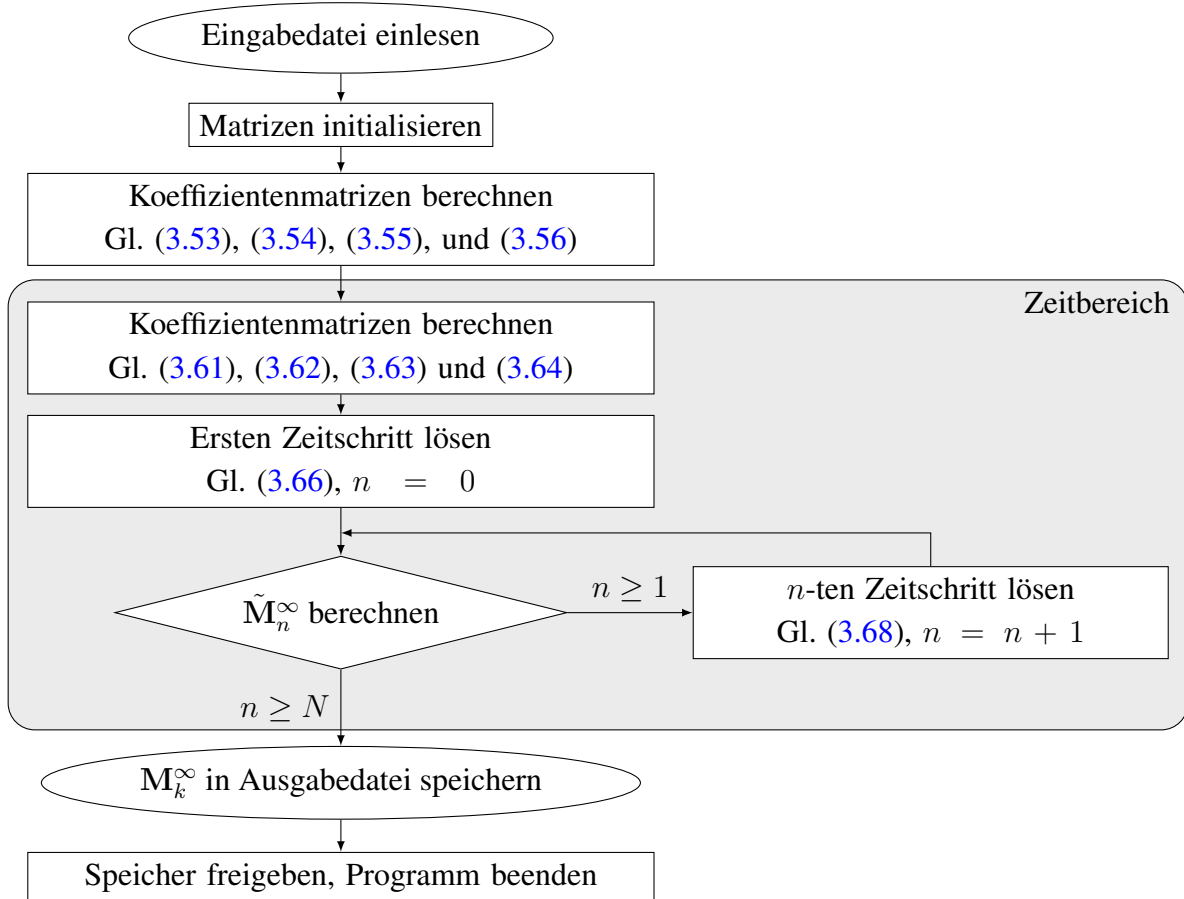


Bild 4.2.: Flussdiagramm der entscheidenden Schritte des Programmablaufs von SCABO.

Der Programmablauf von SCABO ist in Bild 4.2 schematisch dargestellt. Zu Beginn der Simulation werden die Eingabedatei und die Steuerbefehle von allen n Prozessen gleichzeitig eingelesen, sodass jeder einzelne Prozess alle Informationen der aktuellen Problemstellung kennt. Anschließend erfolgt die Ermittlung der Freiheitsgrade jedes einzelnen Knotens zur Bestimmung der Systemgröße. Bis hier entspricht der Programmablauf einem n -Mal gestarteten sequentiellen Programm, da die vorliegenden Daten auf allen n Prozessen gleich sind. Ist die Systemgröße bekannt, kann der Speicher für das globale System zugewiesen werden und der parallele Programmablauf beginnt. Jeder einzelne Prozess bekommt einen lokalen Teil des globalen Systems zugewiesen und die Elemente werden auf die Prozesse verteilt. Damit unterscheiden sich die Daten auf den einzelnen Prozessen (wie bereits in Abschnitt 4.1 erläutert). Im nächsten Schritt erfolgt die Berechnung der nun lokal vorliegenden Elementkoeffizientenmatrizen, somit hat jeder Prozess nur einen Teil der Elementmatrizen zu berechnen. Die lokalen Elementmatrizen werden in das globale System eingesetzt. Ist dies geschehen, kann mit der Simulation im Zeitbereich begonnen werden. Die Koeffizientenmatrizen aus Gleichung (3.53) bis (3.56) werden in die Koeffizientenmatrizen der Gleichungen (3.61) bis (3.64) überführt. Im ersten Zeitschritt

wird M_0^∞ durch Lösen der Riccati Gleichung (3.66) bestimmt (vgl. [22, 23]). Alle weiteren Einflussmatrizen M_n^∞ werden innerhalb einer Schleife über alle Zeitschritte N berechnet. Sobald die Anzahl der zu bestimmenden Zeitschritte erreicht ist, werden die Einflussmatrizen gespeichert und können für die gekoppelte Simulation genutzt werden. Abschließend wird der Arbeitsspeicher freigegeben und die Anwendung beendet.

4.2.2. *elementary Parallel Solver* ELPASO

Die auf der Finite Elemente Methode basierende Software ELPASO *elementary Parallel Solver* ist ein entwickelter Forschungscode, der auf dem EVA-Prinzip basiert [17]. Neben der in dieser Arbeit untersuchten FEM-SBFEM-Kopplung wird die Software bereits zum Lösen unterschiedlicher Fragestellungen wie beispielsweise Modal- und Strukturanalysen an Steuerungsgeräten von Satelliten sowie im Bereich der Modellierung von akustischen Problemen erfolgreich eingesetzt (vgl. CLASEN [43], BECK et al. [16, 18], WULKAU et al. [166]). Des Weiteren kann die Software auch zur Eigenwertbestimmung und geometrischen Optimierung benutzt werden.

Über eine Eingabedatei, die Interfaceinformationen und das FE-Netz beinhaltet sowie Steuerbefehle wird ELPASO vom Anwender über die Eingabeaufforderung oder auch Terminal bzw. Konsole konfiguriert und gestartet. Zusätzlich zum FE-Netz, bestehend aus Knoten und Elementen, sind in der Eingabedatei auch Informationen bezüglich der Materialien, Randbedingungen und angreifenden Kräfte abgelegt. Die gekoppelten Nahfeld-Fernfeld Simulationen wurden mit der Version 387 durchgeführt [17].

Um die eingangs beschriebenen Vorteile zu nutzen, baut auch ELPASO auf externen Bibliotheken auf. Die wichtigsten sind in Bild 4.3 samt Abstraktionslevel dargestellt. Wie bereits in Bild 4.1 trennt auch hier die obere gestrichelte Linie die Bibliotheken, die speziell für vollbesetzte beziehungsweise dünnbesetzte Matrizen entwickelt wurden und auch die untere gestrichelte Linie grenzt wieder die sequentiellen von den parallel arbeitenden Bibliotheken ab. Anders als bei SCABO basiert ELPASO im Wesentlichen auf PETSc, da aus der Finite Elemente Methode bei entsprechender Optimierung der Netze schmalbandige und somit dünnbesetzte Matrizen resultieren [45].

PETSc stellt neben den wichtigen Datenstrukturen wie beispielsweise parallelen Vektoren und Matrizen auch parallele iterative Gleichungslöser und passende Vorkonditionierer für dünnbesetzte Gleichungssysteme zur Verfügung. Zu den bereitgestellten parallelen iterativen Gleichungslösern gehören unter anderem *Richardson*, *conjugate gradients*, *GMRES* und weitere. Unter den Vorkonditionierern sind beispielsweise *block Jacobi*, die unvollständige *LU*-Zerlegung sowie die ebenfalls unvollständige *Cholesky*-Zerlegung zu finden [7]. Dadurch eignet sich diese Bibliothek besonders gut zum Lösen von Gleichungssystemen, wie sie bei der Finite Elemente Methode auftreten. Die parallele Effizienz dieser Bibliothek wird von den Entwicklern in [9] ausführlich diskutiert. Zwei weitere Bibliotheken, *MULTifrontal Massively Parallel sparse direct*

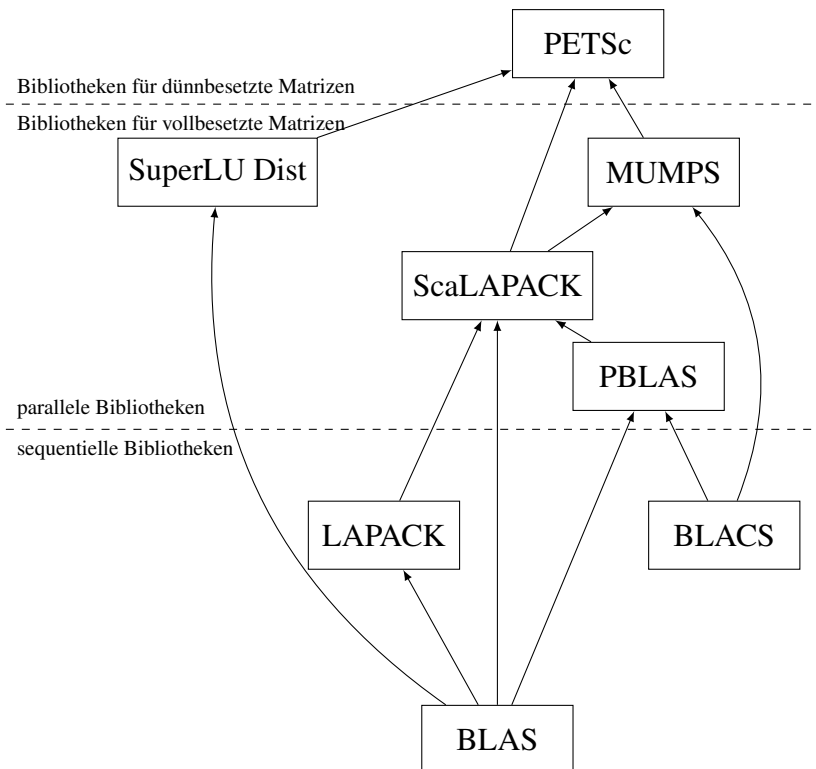


Bild 4.3.: In ELPASO eingebundene Bibliotheken.

Solver (MUMPS) [118] und *SuperLU Dist* [103], eine spezielle Version der Bibliothek *SuperLU* [102, 104] für die Anwendung auf Systemen mit verteiltem Arbeitsspeicher, sind in ELPASO eingebettet, sodass auch direkte Gleichungslöser genutzt werden können. Der Anwender kann somit je nach Problemstellung und Problemgröße einen passenden Gleichungslöser auswählen. Die Basis bilden auch in diesem Fall die zuvor bereits erwähnten Bibliotheken BLAS, LAPACK, PBLAS und BLACS.

4.2.3. FEM-SBFEM Schnittstelle

Die Kopplung von FEM und SBFEM erfolgt über eine definierte Schnittstelle, diese ist in der Lage unterschiedliche Methoden miteinander zu koppeln und Informationen zwischen verschiedenen Programmen auszutauschen. Übergeordnet organisiert der so genannte *Interface Handler* die Kommunikation zwischen den beteiligten Programmen. Der *Interface Handler* kann mehrere Schnittstellen gleichzeitig verwalten, sodass innerhalb einer Simulation mehrere Schnittstellen genutzt werden können. Diese Möglichkeit wird zum Beispiel bei der Nutzung von Substrukturen verwendet. Neben der hier diskutierten FEM-SBFEM Schnittstelle, die bereits für unterschiedliche Fragestellungen eingesetzt wird [138, 139, 141], wurden weitere Schnittstellen entwickelt und in ELPASO implementiert. Zu diesen zählen beispielsweise eine FEM-BEM Schnittstelle für zweidimensionale elastodynamische Fragestellungen im

Zeitbereich [134], FEM-BEM für zweidimensionale und dreidimensionale akustische Untersuchungen im Frequenzbereich [19, 20] oder auch eine Schnittstelle zur Simulation von zweidimensionaler und dreidimensionaler Fluid-Struktur-Interaktion [60, 61] mittels *ELPASO* und *PETSc-FEM* [151].

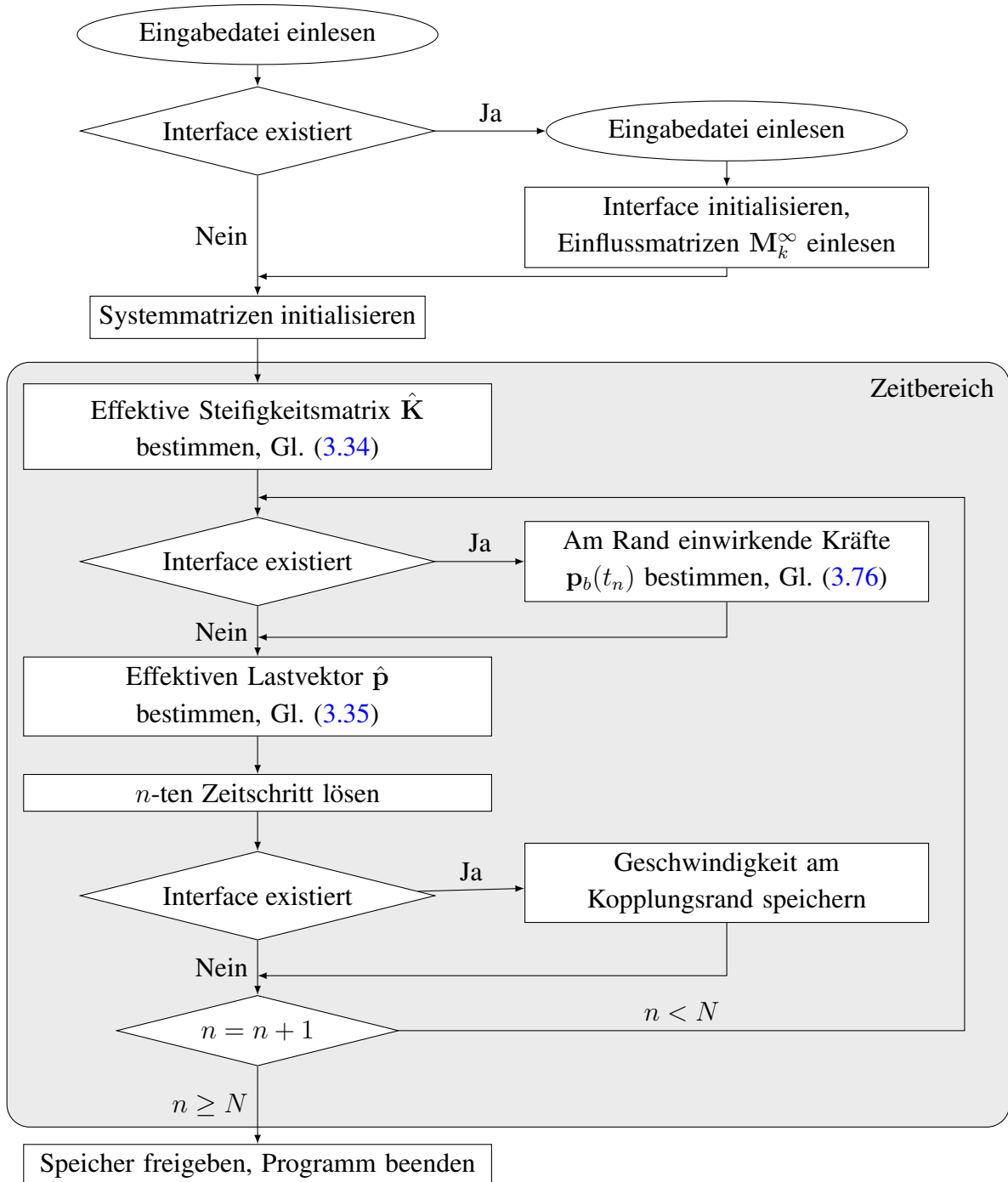


Bild 4.4.: Flussdiagramm des Programmablaufs von *ELPASO* bei der gekoppelten Nahfeld-Fernfeld Simulation.

Der Programmablauf von ELPASO ist für die gekoppelte FEM-SBFEM Simulation als Flussdiagramm in Bild 4.4 dargestellt. Wie bereits in Abschnitt 3.4 erwähnt, handelt es sich um eine bidirektionale Kopplung der beiden Teilsysteme. Die Informationen zur Kopplung von Nahfeld (FEM) und Fernfeld (SBFEM) werden über eine gemeinsame Schnittstelle ausgetauscht. Das Nahfeld übergibt Geschwindigkeiten und Beschleunigungen an das Fernfeld. Das Fernfeld hingegen übergibt Kräfte an das gekoppelte Nahfeld.

Zu Beginn der Simulation werden die Eingabedatei und die Steuerbefehle von allen n Prozessen gleichzeitig eingelesen, sodass jeder einzelne Prozess alle Informationen der aktuellen Problemstellung kennt. Zusätzlich werden die Eingabedateien aus der vorangegangenen Fernfeldberechnung gelesen und die Netzinformationen für die Kopplung geprüft. Die Konnektivität der FEM und SBFEM Knoten wird auf Grundlage ihrer geometrischen Position getestet, dabei muss für jeden SBFEM Knoten ein Kopplungsknoten im Nahfeld verfügbar sein. Ist dies nicht der Fall, folgt ein vorzeitiger Programmabbruch. Ist die Prüfung erfolgreich, beginnen die Prozesse die Einflussmatrizen M_k^∞ einzulesen. Das Lesen der Matrizen erfolgt verteilt, sodass jeder Prozess nur einen Teil jeder Einflussmatrix speichert.

Anschließend werden von allen Prozessen die benötigten Freiheitsgrade an jedem einzelnen Knoten für das gesamte Netz ermittelt und so die Systemgröße bestimmt. Ist die Systemgröße bekannt, kann der Speicher für das globale System zugewiesen werden und der parallele Programmablauf beginnt. Systemmatrizen und Systemvektoren werden erstellt, dabei bekommt jeder einzelne Prozess einen lokalen Teil des globalen Systems zugewiesen. Die Elemente werden auf die Prozesse verteilt, sodass jeder Prozess in etwa die gleiche Anzahl an Elementmatrizen auszuwerten hat. Im nächsten Schritt erfolgt die Berechnung der nun lokal vorliegenden Elementsteifigkeitsmatrizen und Elementmassenmatrizen sowie der Elementlastvektoren (vgl. Gleichungen (3.7), (3.9) und (3.10)). Die lokalen Matrizen und Vektoren werden in die globalen Matrizen M , K und den globalen Lastvektor p eingesetzt. Sind alle lokalen Matrizen und Vektoren in globale Matrizen und Vektoren eingesetzt, ist das System vollständig beschrieben. Die Dämpfungsmatrix C wird dann wie in Gleichung (3.12) beschrieben bestimmt. Ist dies geschehen und die Randbedingungen sowie die Anfangswertbedingungen der Verschiebung, Geschwindigkeit und Beschleunigung nach Gleichungen (3.16) und (3.17) gesetzt, folgt die Berechnung im Zeitbereich (vgl. grau hinterlegten Kasten in Bild 4.4).

Zu Beginn der Simulation wird die effektive Steifigkeitsmatrix \hat{K} (vgl. Gleichung (3.34)) bestimmt. Dies ist bei linearen Systemen konstanter Steifigkeit, Dämpfung und Masse nur im ersten Schritt notwendig. Danach wird die Schleife über alle Zeitschritte durchlaufen. Die Kopplungskräfte am Interface werden bestimmt und zu dem Elementlastvektor hinzugefügt. Sollte das Fernfeld in mehrere Substrukturen aufgeteilt sein, erfolgt dieser Schritt für jedes einzelne Interface. Sind die angreifenden Kräfte bekannt, kann der effektive Lastvektor \hat{p} nach Gleichung (3.35) ermittelt werden. Der n -te Zeitschritt wird gelöst und die Anfangswertbedingungen gemäß der Gleichungen (3.19) und (3.31) für den nächsten Schritt gesetzt. Dem Interface werden die aktuellen Geschwindigkeiten der FE-Knoten übergeben, um die zur Berechnung der Kräfte

des Fernfeldes nötigen Geschwindigkeiten im Interface zu speichern. Ist die maximale Anzahl an Zeitschritten erreicht, wird der Speicher freigegeben und das Programm beendet, andernfalls folgt die Berechnung des nächste Zeitschritts.

5. Validierung des gekoppelten Ansatzes

In diesem Kapitel wird gezeigt, dass der im Kapitel 3.4 beschriebene gekoppelte Lösungsansatz aus FEM und SBFEM die Sommerfeld'sche Abstrahlbedingung (vgl. Kapitel 2.4) erfüllt. Anhand zweier ausgewählter Beispiele wird das System validiert. Zusätzlich erfolgt eine nähere Betrachtung des Einflusses des Zeitintegrationsverfahrens auf das Ergebnis der numerischen Berechnung.

Das erste Beispiel verdeutlicht die Wirkungsweise der SBFEM und zeigt das unterschiedliche Verhalten brandeter und unbrandeter Gebiete. Bei diesem Beispiel wird der Einfluss der Parameterwahl des Zeitintegrationsverfahrens auf die Lösung deutlich aufgezeigt und diskutiert.

Das zweite Beispiel ist eine Setzungssimulation, an Hand derer eine Konvergenzstudie bezüglich des sich einstellenden Fehlers durchgeführt wird. Dabei wird die Anzahl der Systemfreiheitsgrade sukzessive erhöht und mit einer analytischen Lösung verglichen.

5.1. Balken

Anhand dieses einfachen Beispiels wird die Komplexität numerischer Simulationen im Zeitbereich aufgezeigt. In der Regel gilt es, unterschiedlichsten Anforderungen gerecht zu werden, die sich gegebenenfalls nicht vereinbaren lassen oder gar ausschließen.

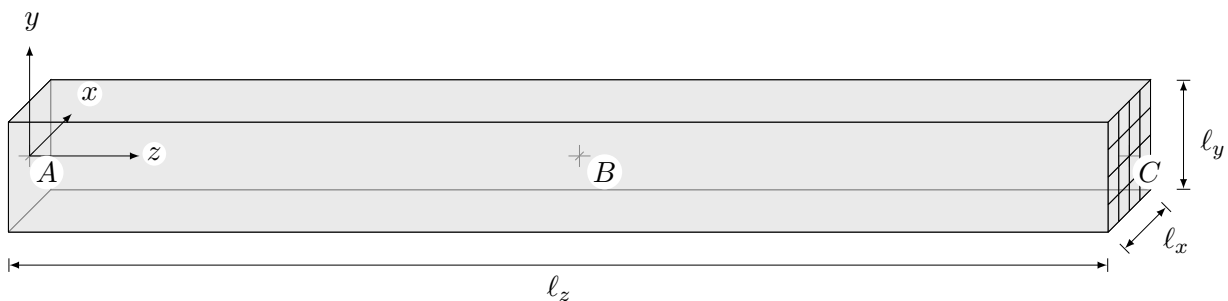


Bild 5.1.: Systemskizze des Nahfeldes der Dimension ℓ_x , ℓ_y und ℓ_z :

Am linken Rand A wird der Impuls (vgl. Bild 5.2) aufgegeben.

Der rechte Rand C ist fest eingespannt oder mit dem Fernfeld gekoppelt.

Es sei ein Volumen (Nahfeld), wie in Bild 5.1 dargestellt der Länge $\ell_z = 1,0$ m, Höhe $\ell_y = 0,1$ m und Tiefe $\ell_x = 0,1$ m definiert und mittels Kontinuums-elementen diskretisiert, sodass jedes einzelne Element eine Kantenlänge von 2,5 cm aufweist. Diesen Elementen wird ein isotropes linear elastisches Material zugewiesen. Die Materialparameter sind wie folgt definiert: $E = 300,0$ MNm⁻², $\nu = 0,15$ und $\rho = 2446,0$ kgm⁻³. Aus den Materialparametern resultiert eine p -Wellengeschwindigkeit $c_p = 359,87$ ms⁻¹, für die Zeitschrittlänge wird $\Delta t = 1 \cdot 10^{-5}$ s gewählt, somit benötigt die p -Welle rund sieben Rechenschritte von einem Element zum nächsten und kann verlustfrei und vollständig abgebildet werden. Die Simulation läuft über 5000 Zeitschritte und somit 0,05 s.

Rechtsseitig ist das Gebiet entweder fest eingespannt oder mit dem Fernfeld gekoppelt. Alle anderen Seiten sind frei. Linksseitig wird am freien Ende auf der gesamten Fläche des Gebiets ein Impuls in Form eines Ricker *Wavelets* aufgebracht. Jeder Knoten ist durch die zeitveränderliche Last

$$R(t) = \left(r_1 + 2\pi r_2 \left(\frac{t - r_4}{r_3} \right)^2 \right) e^{-2\pi \left(\frac{t - r_4}{r_3} \right)^2} \quad (5.1)$$

senkrecht zur Fläche belastet. Das *Wavelet* ist derart formuliert, dass Amplitude, Breite und Länge des Impulses über die Parameter r_1 , r_2 und r_3 definiert werden können. Der Zeitpunkt der maximalen Amplitude wird über den Parameter r_4 bestimmt. Auf Eckknoten und Kantenknoten ist jeweils ein Viertel beziehungsweise die Hälfte der Lastfunktion aus Gleichung (5.1) aufgebracht.

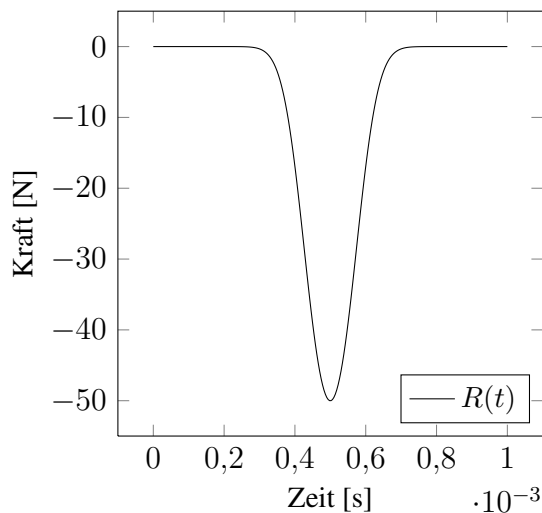


Bild 5.2.: Funktionsverlauf der aufgeprägten Lastfunktion im Zeitbereich.

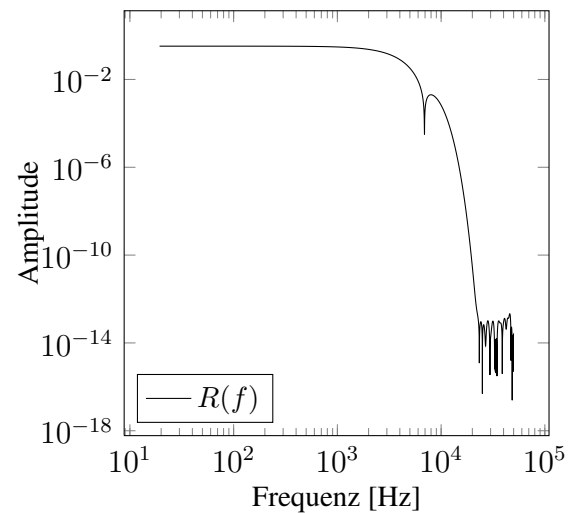


Bild 5.3.: Darstellung der Lastfunktion im Frequenzbereich.

Bild 5.2 zeigt den zeitabhängigen Funktionsverlauf der Gleichung (5.1) mit $r_1 = -50$, $r_2 = -20$, $r_3 = 0,0002$ und $r_4 = 0,0005$. Mittels schneller Fourier-Transformation FFT (*Fast Fourier Transformation*) kann die gegebene Funktion vom Zeitbereich in den Frequenzbereich und umgekehrt übertragen werden. Ausführliche Informationen zur Fourier Transformation sowie der Funktionsweise der FFT sind beispielsweise [101] bzw. [39, 55] oder auch [75] zu entnehmen. Die durch den Impuls angeregten Frequenzen sind in Bild 5.3 dargestellt. Es wird deutlich, dass der Impuls ein breites Band von Frequenzen anregt. Im Bereich von circa 20 bis 500 Hz ist die Amplitude der einzelnen Frequenzen nahezu konstant, danach nimmt die Amplitude mit steigender Frequenz stetig ab. Ab einer Frequenz von 8000 Hz fällt die Amplitude der nachfolgenden Frequenzen deutlich.

Drei Punkte werden gewählt, um die Simulation auszuwerten und die Auswirkungen der Wahl der Zeitintegrationsparameter zu beurteilen. Knoten A liegt in der Mitte des linken Randes,

Punkt	(x, y, z)	Randbedingung
<i>A</i>	($\ell_x/2$, $\ell_y/2$, 0)	Frei
<i>B</i>	($\ell_x/2$, $\ell_y/2$, $\ell_z/2$)	-
<i>C</i>	($\ell_x/2$, $\ell_y/2$, ℓ_z)	Festgehalten bzw. ans Fernfeld gekoppelt

Tabelle 5.1.: Koordinaten und Randbedingung der Auswertungspunkte *A*, *B* und *C*.

Knoten *B* im Zentrum des Nahfeldes und Knoten *C* in der Mitte des rechten Randes. Die exakten geometrischen Koordinaten sind in Tabelle 5.1 zusammengefasst. Werden am rechten Rand die Knoten fest eingespannt, sind dort die Knotenfreiwerte der Verschiebung zu jedem Zeitpunkt exakt gleich Null. Dasselbe gilt ebenfalls für die Geschwindigkeit und die Beschleunigung.

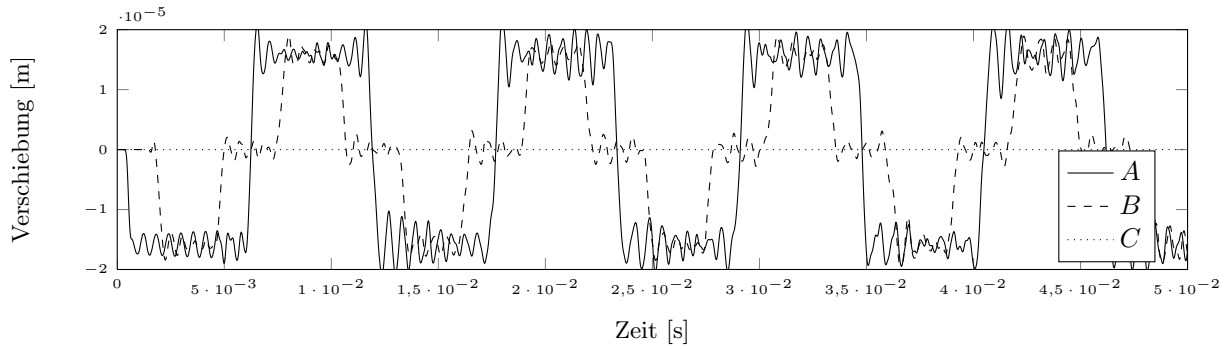
5.1.1. Numerische Lösung

Die Ergebnisse der numerischen Simulation hängen direkt von dem verwendeten Zeitintegrationsverfahren ab. Für unterschiedliche Zeitintegrationsverfahren sind die Ergebnisse der Berechnung in den Bildern 5.4 und 5.5 zusammengefasst. Sie zeigen die zeitabhängige Verschiebung in *z*-Richtung der drei Referenzknoten (*A*, *B* und *C*) sowie die Energie des Gesamtsystems, die mit dem jeweiligen Zeitintegrationsverfahren bestimmt wurden. Die Verschiebungen in *x*- und *y*-Richtung sind sehr klein ($< 1,0 \cdot 10^{-15}$) und werden nicht weiter untersucht.

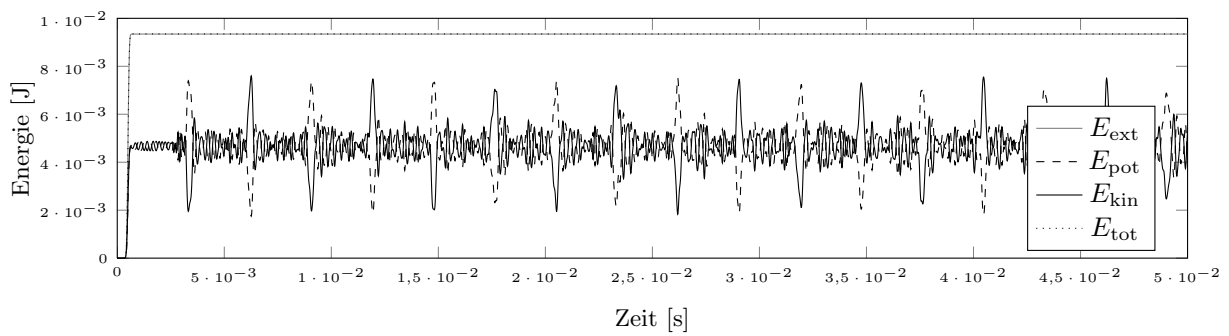
Bild 5.4a zeigt für die drei Knoten *A*, *B* und *C* den Zeit-Verschiebungsverlauf, der mittels Zeitschrittintegrationsverfahren (vgl. Abschnitt 3.2 Newmark-Parameter $\beta = 0,25$ und $\gamma = 0,5$) bestimmt ist. Aufgrund des eingeleiteten Impulses wird ein breites Spektrum an Frequenzen angeregt, wodurch die Grundschwingung des Systems von weiteren Schwingungen unterschiedlichster Frequenzen überlagert wird. Die Verschiebung des Knoten *C* ist, wie zu erwarten, exakt gleich Null, da dieser Knoten während der gesamten Berechnung festgehalten wird. Aufgrund des aufgeprägten Impulses läuft eine Kompressionswelle in Richtung der *z*-Achse durch das Gebiet. Diese Welle wird zum Zeitpunkt

$$T = \ell_z / c_p, \quad (5.2)$$

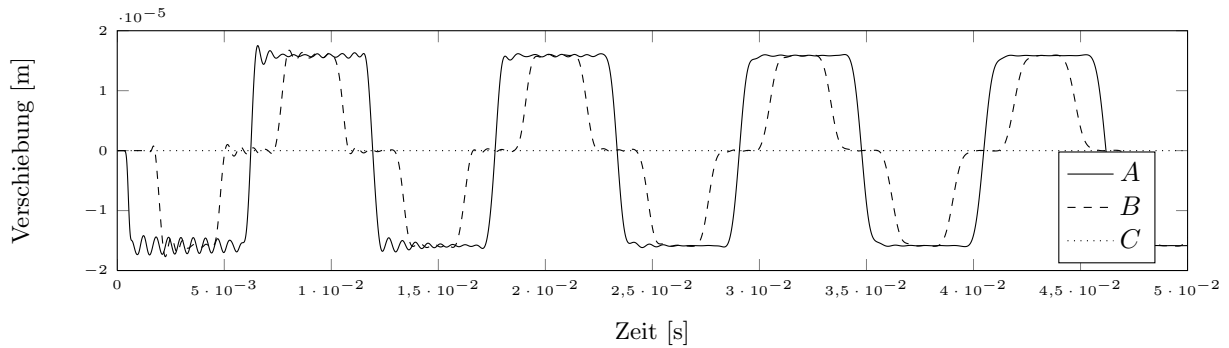
nachdem sie den Knoten *B* bei $T/2$ passiert, am rechten Rand ($\mathbf{u} = \dot{\mathbf{u}} = \ddot{\mathbf{u}} = 0,0$) vollständig reflektiert und läuft in entgegengesetzter Richtung zurück. Zum Zeitpunkt $2T$ erreicht die Welle dann den linken Rand und wird auch dort vollständig reflektiert. Da es sich bei dem linken Rand um ein freies Ende handelt, sind Verschiebung \mathbf{u} , Geschwindigkeit $\dot{\mathbf{u}}$ und Beschleunigung $\ddot{\mathbf{u}}$ ungleich Null. Es resultiert eine Phasenumkehr. In Bild 5.4b sind vier unterschiedliche Energieverläufe dargestellt. Der Graph E_{ext} beschreibt die durch den Impuls von außen ins System eingebrachte Energie (vgl. Gleichung (3.48)), mit Aufbringen des Impulses steigt die Energie an und bleibt bis zum Ende der Simulation konstant, da keine weiteren Kräfte auf das



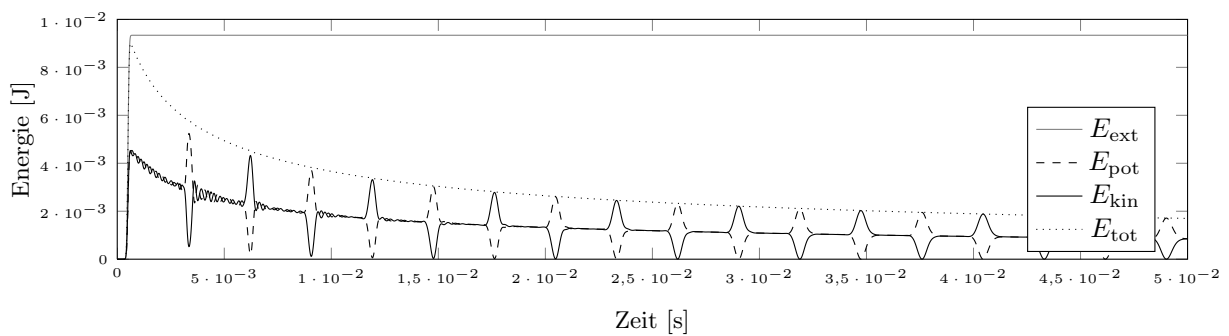
(a) Zeit-Verschiebungsverlauf der FEM Simulation.



(b) Energieverlauf der FEM Simulation.



(c) Zeit-Verschiebungsverlauf der FEM Simulation.



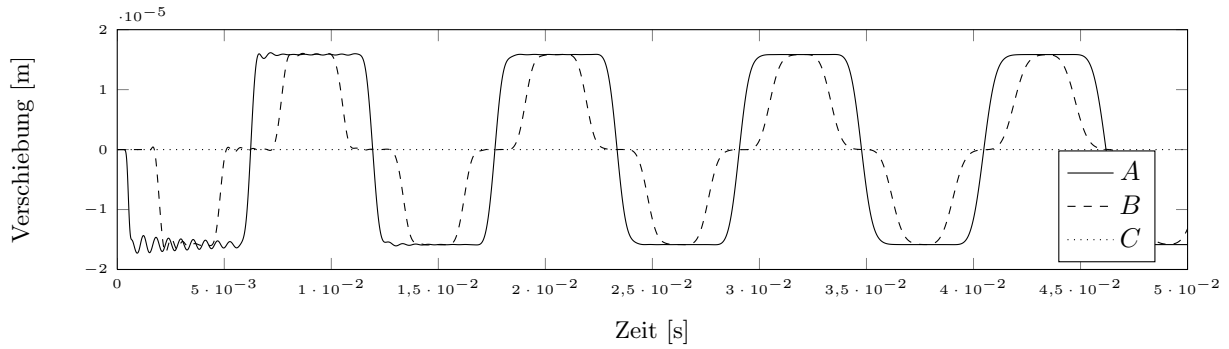
(d) Energieverlauf der FEM Simulation.

Bild 5.4.: Simulation des Nahfeldes

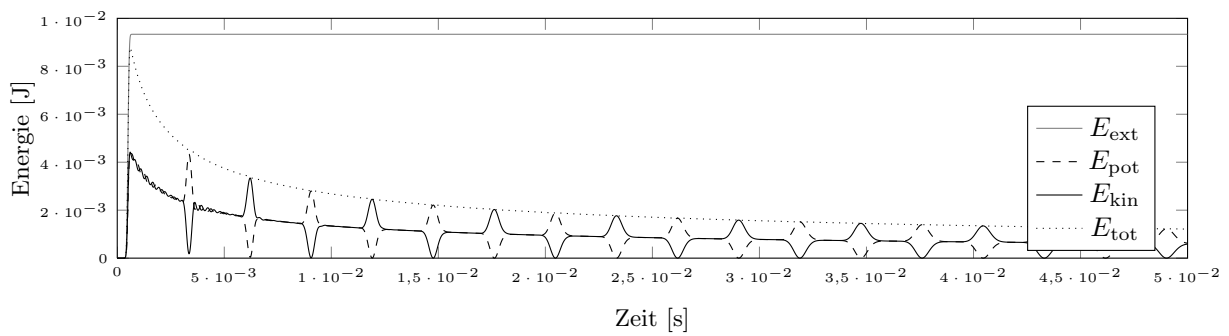
ohne numerische Dämpfung (a und b) mit den Newmark: $\beta = 0,25$ und $\gamma = 0,5$

und mit numerischer Dämpfung (c und d) mit den Newmark: $\beta = 0,390625$ und $\gamma = 0,75$.

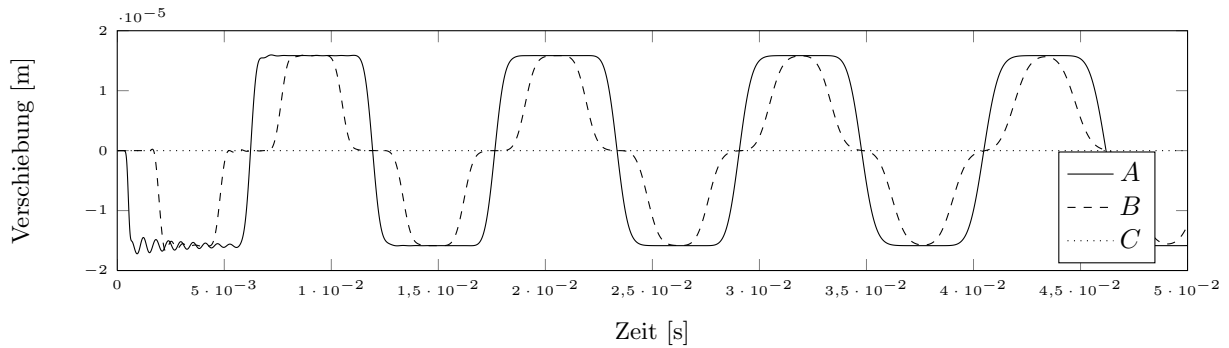
5 Validierung des gekoppelten Ansatzes



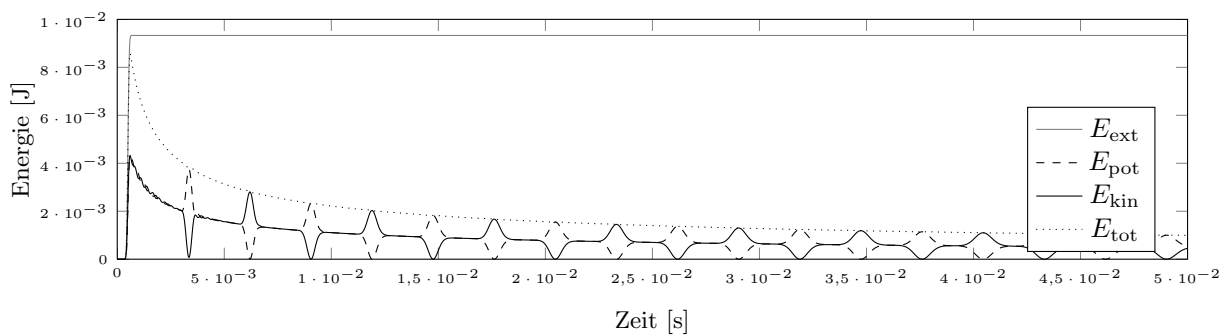
(a) Zeit-Verschiebungsverlauf der FEM Simulation.



(b) Energieverlauf der FEM Simulation.



(c) Zeit-Verschiebungsverlauf der FEM Simulation.



(d) Energieverlauf der FEM Simulation.

Bild 5.5.: Simulation des Nahfeldes

mit numerischer Dämpfung mit den HHT- α : $\alpha = -0,25$, $\beta = 0,390625$ und $\gamma = 0,75$ (a und b)
sowie den Generalized- α : $\alpha_m = 0,125$, $\alpha_f = 0,375$, $\beta = 0,390625$ und $\gamma = 0,75$ (c und d).

System einwirken. E_{pot} und E_{kin} beschreiben die potentielle beziehungsweise kinetische Energie des Systems (vgl. Gleichung (3.47)). Die Summe der beiden Energien ist im Graphen E_{tot} zusammengefasst. Dieser überdeckt während der gesamten Simulation den Graphen E_{ext} . Dies bedeutet, dass die Energie im System exakt der von außen aufgebrauchten Energie entspricht. Es gibt somit keinen Energieverlust aufgrund von Dämpfung, das numerische Zeitschrittintegrationsverfahren ist nicht dissipativ, weshalb von einem mechanisch und numerisch konservativen System gesprochen wird.

Wie bereits in Kapitel 3.2 diskutiert, ist es möglich, eine numerische, nicht physikalische Dämpfung mittels der Zeitintegrationsparameter zu realisieren. Im Newmark-Verfahren lässt sich diese über die Konstante γ regeln. Ist $\gamma > 0,5$ gewählt, erfährt das System eine numerische Dämpfung. In diesem Fall handelt es sich dann um ein dissipatives Verfahren. Der zeitliche Verlauf der Verschiebungen der drei Referenzknoten ist in Bild 5.4c für $\beta = 0,390625$ und $\gamma = 0,75$ dargestellt. Der Verschiebungsverlauf fällt hier verglichen mit dem in Bild 5.4a deutlich glatter aus. Die überlagernden Schwingungen werden merklich gedämpft, sind jedoch auch nach $10T$ deutlich zu erkennen. Der Energieverlauf in Bild 5.4d verdeutlicht die Wirkung der eingebrachten Dämpfung. E_{ext} beschreibt auch hier die ins System eingebrachte Energie und entspricht der Energie im vorangegangenen Beispiel. In diesem Fall überdeckt der Graph E_{tot} den Graphen E_{ext} nicht mehr, da ein Teil der Energie dem System über die Zeit entzogen wird. Sobald die kinetische Energie vollständig aus dem System verschwunden ist, befindet sich das System in seiner statischen Ruhelage. In diesem Fall ist die potentielle Energie konstant und entspricht der Lageenergie. Zum Ende der Simulation nach $5 \cdot 10^{-2}$ s sind lediglich 18,2% der durch den Impuls eingebrachten Energie übrig.

Zusätzlich zu der Konstante γ können weitere Parameter eingeführt werden (vgl. Abschnitt 3.2.2 und 3.2.3). Diese zusätzlich eingeführten Parameter (α , α_m und α_f) dämpfen vorrangig hochfrequente Schwingungen. Die Bilder 5.5a und 5.5c zeigen die Zeit-Verschiebungsverläufe einer Simulation mit dem HHT- α -Verfahren und den Parametern $\alpha = -0,25$, $\beta = 0,390625$ und $\gamma = 0,75$ und dem Generalized- α -Verfahren mit den Parametern $\alpha_m = 0,125$, $\alpha_f = 0,375$, $\beta = 0,390625$ und $\gamma = 0,75$ ($p_\infty = 0,6$). Die hochfrequenten Schwingungen sind beim HHT- α -Verfahren nach $6T$ nahezu vollständig abgebaut. Beim Generalized- α -Verfahren ist dies schon nach $4T$ der Fall. Diese zusätzliche Dämpfung sorgt für eine noch schnellere Reduktion der Systemenergien wie in den Bildern 5.5b und 5.5d dargestellt. Nach $5 \cdot 10^{-2}$ s Simulationszeit liegt die Systemenergie im ersten Fall bei lediglich 12,98% der eingetragenen Energie. Im zweiten Fall liegt die Systemenergie bei 10,63% und ist somit noch weiter reduziert.

Der Dämpfungseffekt lässt sich in einer anderen Darstellung besser aufzeigen. Dazu werden die Verschiebungen des Referenzknotens aus dem Zeitbereich in den Frequenzbereich mittels diskreter FFT transformiert. Exemplarisch sei dies für den Referenzknoten A gezeigt. In diesem Fall ist die resultierende Auflösung der Frequenzen mit $f = \frac{1}{5000\Delta t} = 20$ Hz relativ grob, genügt jedoch, um den Dämpfungseffekt der einzelnen Zeitschrittverfahren zu veranschaulichen.

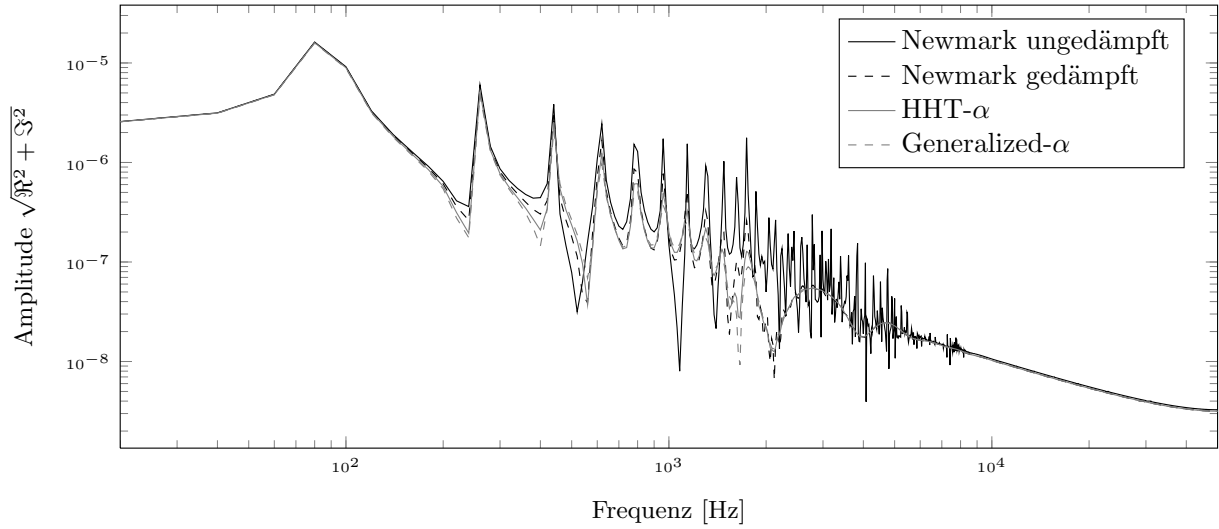


Bild 5.6.: Darstellung der Verschiebung des Referenzknotens A im Frequenzbereich.

Die Dämpfung der höheren Frequenzen ist aufgrund ihrer kleineren Amplitude deutlich zu erkennen.

Bild 5.6 zeigt die Verteilung der Amplituden je Frequenz in 20 Hz Schritten in einer doppelt logarithmischen Darstellung. Die sichtbare Dämpfung der Frequenzen beginnt bei etwa 120 Hz und nimmt mit steigender Frequenz zu. Je größer die Differenz der Amplituden zwischen der ungedämpften Simulation und den gedämpften Simulationen ausfällt, desto größer ist die Dämpfung des gewählten Zeitschrittverfahrens. Damit entspricht diese Form der Darstellung den Erkenntnissen aus der Betrachtung der Systemenergien (vgl. Bild 5.4b, 5.4d, 5.5b und 5.5d).

Je nach Anwendungsfall ist ein dazu passendes Zeitschrittintegrationsverfahren zu wählen. Unabhängig davon ist festzuhalten, dass alle hier diskutierten Zeitintegrationsverfahren für die Kopplung von FEM und SBFEM geeignet sind.

5.1.2. Analytische Lösung

Der in diesem Beispiel aufgebraachte Impuls führt näherungsweise zu einer Rechteckschwingung des modellierten Körpers im Punkt A , eine analytisch Lösung kann mittels Fourier Reihe angenähert werden. Die allgemeine Form der Fourier Reihe ist wie folgt beschrieben [11]

$$s(t) = a_0 + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi kt}{T}\right) + \sum_{k=1}^{\infty} b_k \sin\left(\frac{2\pi kt}{T}\right) \quad (5.3)$$

und muss für die Darstellung der Rechteckfunktion entsprechend angepasst werden. Dabei sind a_k und b_k die Amplituden der Teilschwingungen, aus denen sich die Gesamtschwingung zusammensetzt. Diese Amplituden werden mit dem nachfolgenden Ansatz zur Beschreibung der

Rechteckfunktion bestimmt

$$\begin{pmatrix} a_k \\ b_k \end{pmatrix} = \frac{2}{T} \left(\int_0^{\frac{T}{2}} \begin{pmatrix} \cos\left(\frac{2\pi kt}{T}\right) \\ \sin\left(\frac{2\pi kt}{T}\right) \end{pmatrix} dt - \int_{\frac{T}{2}}^T \begin{pmatrix} \cos\left(\frac{2\pi kt}{T}\right) \\ \sin\left(\frac{2\pi kt}{T}\right) \end{pmatrix} dt \right). \quad (5.4)$$

Aus der Auswertung resultiert, dass die Koeffizienten a_k der Kosinus-Anteile alle gleich Null sind. Die Koeffizienten b_k der Sinus-Anteile hingegen ergeben sich wie folgt:

$$b_k = \begin{cases} \frac{4}{\pi k} & \text{wenn } \{2k+1 : k \in \mathbb{Z}\} \\ 0 & \text{wenn } \{2k : k \in \mathbb{Z}\} \end{cases} \quad (5.5)$$

Somit kann Gleichung (5.3) unter Berücksichtigung lediglich der ungeraden Zahlen für k zu

$$s(t) = \sum_{k=1,3,\dots}^{\infty} \frac{4}{\pi k} \sin\left(\frac{2\pi kt}{T}\right) \quad (5.6)$$

vereinfacht werden.

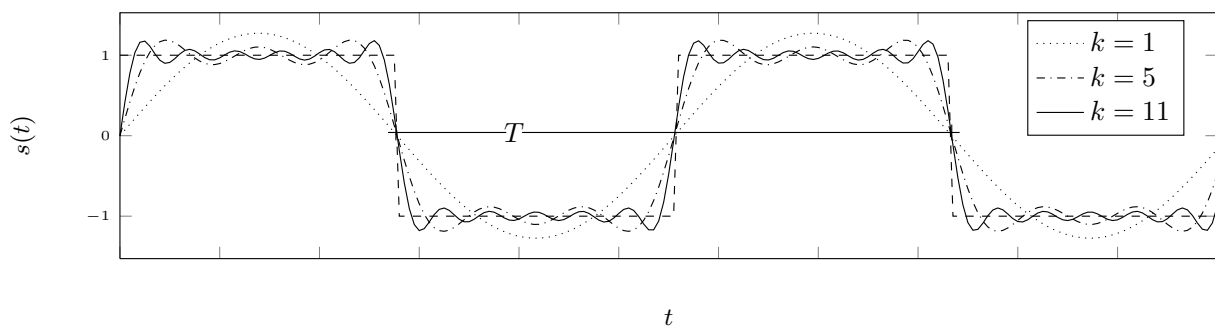


Bild 5.7.: Qualitativer Funktionsverlauf der analytischen Lösung (5.6) für k gleich eins, fünf und elf als Annäherung an die Rechteckschwingung.

Bild 5.7 zeigt die in Gleichung (5.6) aufgestellte Reihe unter Berücksichtigung eines einzelnen Terms der Reihe sowie von fünf bzw. elf Termen. Je höher die Anzahl der ausgewerteten und summierten Terme, desto genauer die Approximation der gesuchten Lösung. Dabei ist jedoch zu beachten, dass jeder der Summe hinzugefügte Sinus mit $\frac{1}{k}$ gewichtet wird, wodurch die Amplitude jedes weiteren Sinus kleiner ist als die des vorherigen und damit der Einfluss jedes weiteren Terms auf die Lösung abnimmt. Es ist somit ein passendes Abbruchkriterium zu wählen. Damit die numerische Lösung mit dem analytischen Ansatz verglichen werden kann, ist der analytische Ansatz dem gegebenen Problem anzupassen. Die Amplitude $a^* = -1,58488 \cdot 10^{-5} \text{ m}$, mit der die analytische Lösung gewichtet wird, ist dem Diagramm 5.5c entnommen. Die Periode $T^* = 4T = \frac{4\ell_z}{c}$ folgt aus der gegebenen Geometrie und der Dehngeschwindigkeit $c = \sqrt{\frac{E}{\rho}}$. Zusätzlich zu den bisherigen Parametern wird ein weiterer Parameter $t_{\text{offset}} = 5,5 \cdot 10^{-4} \text{ s}$ eingeführt, mit dem der zeitliche Versatz der Lösung berücksichtigt werden kann. Dieser ist in der Art gewählt, dass das erste Maximum der analytischen

Lösung auf den Zeitpunkt nach Einwirken des vollständigen Impulses fällt. Somit hat die angepasste Lösung der Gleichung (5.6) die folgende Gestalt:

$$s(t) = a^* \sum_{k=1,3,\dots}^{\infty} \frac{4}{\pi k} \sin\left(\frac{2\pi k (t - t_{\text{offset}})}{T^*}\right) \quad (5.7)$$

$$= -1,58488 \cdot 10^{-5} \sum_{k=1,3,\dots}^{\infty} \frac{4}{\pi k} \sin\left(\frac{350,21\pi k (t - 5,5 \cdot 10^{-4})}{2}\right) \quad (5.8)$$

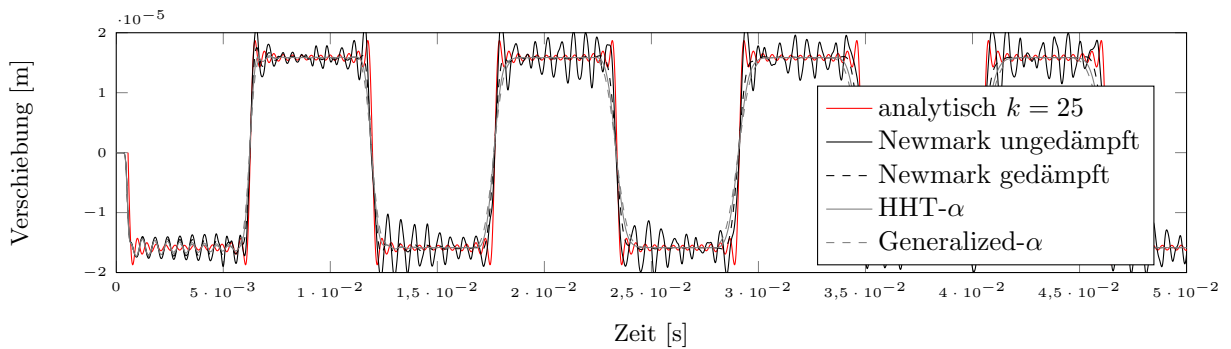


Bild 5.8.: Vergleich der analytischen Lösung mit den numerischen Ergebnissen für den Punkt A.

Bei fortschreitender Simulation bleibt die Amplitude im ungedämpften Fall konstant. Wird eine numerische Dämpfung über das Zeitschrittintegrationsverfahren eingebracht, nimmt die Amplitude, wie bereits diskutiert, ab. Neben der Amplitude wird auch die Periode durch das Zeitintegrationsverfahren bei entsprechender Wahl der Zeitschrittlänge Δt korrekt abgebildet. Ein Periodenfehler ist in dem hier betrachteten Zeitintervall nicht sichtbar. Wird Δt zu groß gewählt, kommt es zu einer Phasenverschiebung, die numerische Periode $T_{\text{numerisch}}$ stimmt dann nicht mehr mit der analytischen Periode $T_{\text{analytisch}}$ überein. Der Phasenfehler wird dann mit $\varphi_{\text{err}} = \frac{T_{\text{numerisch}}}{T_{\text{analytisch}}} - 1$ bestimmt (vgl. z. B. [79]).

5.1.3. Numerische Lösung des gekoppelten Verfahrens

Bei der gekoppelten Simulation von Nah- und Fernfeld werden die Knoten des rechten Randes an ein Fernfeld gleichen Materials gekoppelt (vgl. Bild 5.1). Damit entfällt gleichzeitig die Festhaltung dieser nun gekoppelten Knoten. Die am Kopplungsrand eintreffende Welle soll nicht wie zuvor reflektiert werden, sondern vom Nahfeld vollständig in das Fernfeld übergehen. Da das angekoppelte Fernfeld die Sommerfeld'sche Abstrahlbedingung erfüllt (vgl. Abschnitt 2.4), sollen keine Wellen aus dem angekoppelten Fernfeld zurück ins Nahfeld kommen. Das bedeutet, dass nach dem Eintreffen der Welle am rechten Rand zum Zeitpunkt T (vgl. Gleichung (5.2)) die Welle das Nahfeld in Richtung der z -Achse verlässt und im unendlichen Fernfeld verschwindet. Dabei spielt die Position des Skalierungszentrums eine wichtige Rolle. Aufgrund

der gewählten Formulierung (vgl. Bild 3.2) ist das Fernfeld nicht rechteckig wie das Nahfeld, sondern hat die Form eines Pyramidenstumpfes wie in Bild 5.9 schematisch dargestellt.

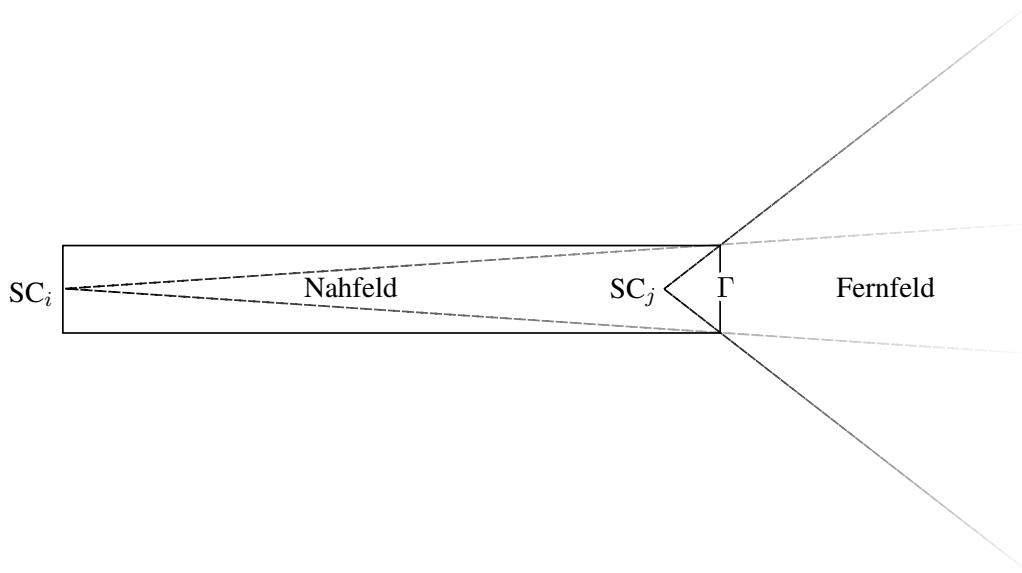


Bild 5.9.: Schematische Darstellung des Nahfeldes aus Bild 5.1 sowie geometrische Veränderung des Fernfeldes in Abhängigkeit der Koordinaten des Skalierungszentrums SC.

Aufgrund der geometrischen Änderung kann es trotz des offenen Randes zu Reflexionen am Interface Γ kommen. Wird vereinfacht von einem eindimensionalen Modell ausgegangen, dann ist die Kraft F , die im Punkt A eingeleitet wird, proportional zur Geschwindigkeit in diesem Punkt. Der Proportionalitätsfaktor Z entspricht dem dynamischen Widerstand, der Impedanz. Es gilt somit

$$F = Zv \quad (5.9)$$

mit

$$Z = A\sqrt{E\rho}. \quad (5.10)$$

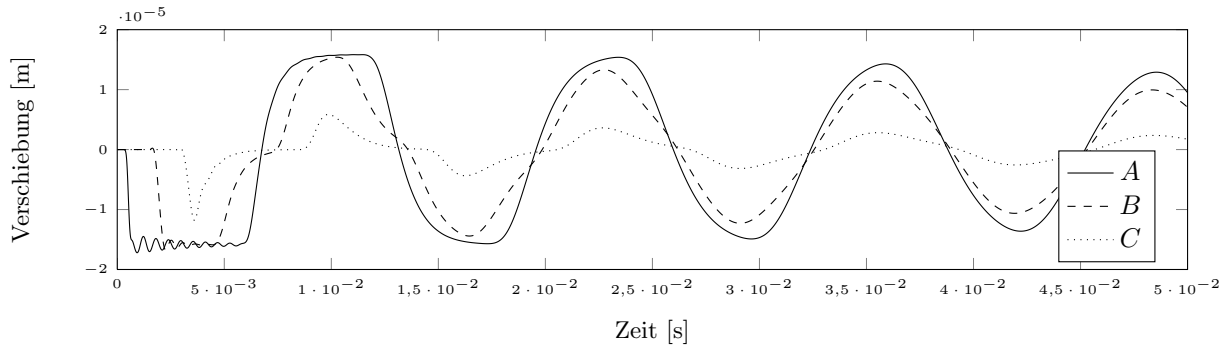
Dabei wird deutlich, dass, obwohl die Materialparameter von Nah- und Fernfeld identisch sind, die Impedanzen aufgrund der Querschnittsveränderung nicht gleich sind. Aufgrund dieses Unterschiedes lassen sich der Reflektionsfaktor

$$r = \frac{Z_1 - Z_2}{Z_1 + Z_2} \quad (5.11)$$

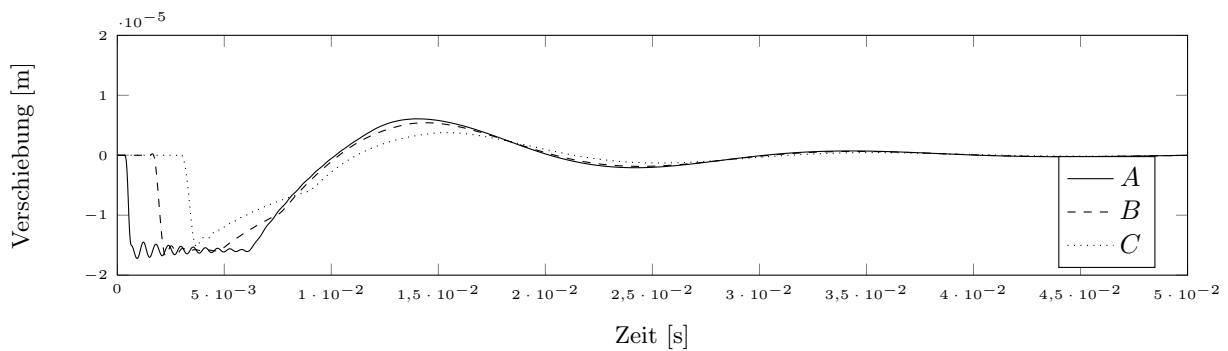
sowie der Transmissionsfaktor

$$t = 1 + r \quad (5.12)$$

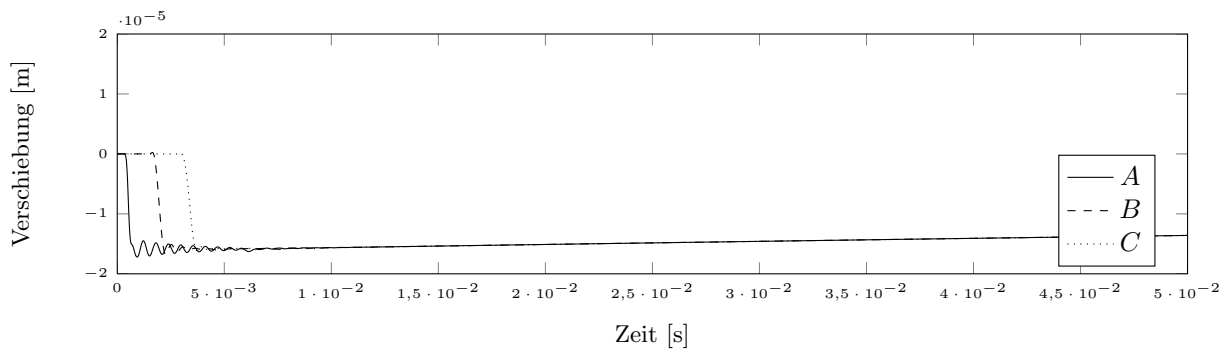
bestimmen. Der Reflektionsfaktor beschreibt den Teil der eintreffenden Welle, der am Rand Γ zurück in das Nahfeld reflektiert wird, der Transmissionsfaktor folglich den Anteil, der durch den Rand verschwindet.



(a) Zeit-Verschiebungsverlauf der gekoppelten FEM/SBFEM Simulation mit Skalierungszentrum SC_1 .



(b) Zeit-Verschiebungsverlauf der gekoppelten FEM/SBFEM Simulation mit Skalierungszentrum SC_2 .



(c) Zeit-Verschiebungsverlauf der gekoppelten FEM/SBFEM Simulation mit Skalierungszentrum SC_3 .

Bild 5.10.: Simulation von Nah- und Fernfeld

mit den Generalized- α : $\alpha_m = 0,125$, $\alpha_f = 0,375$, $\beta = 0,390625$ und $\gamma = 0,75$.

Den zeitlichen Verlauf der Verschiebung für die Knoten A , B und C geben die Bilder 5.10a bis 5.10c für die in der Tabelle 5.2 aufgelisteten Skalierungszentren wieder. Bei dieser Untersuchung sind lediglich die Ergebnisse der Simulationen mit dem Generalized- α -Verfahren dargestellt. Für die anderen zuvor diskutierten Verfahren sind die zeitlichen Verläufe wie zu erwarten sehr ähnlich und werden hier deshalb nicht explizit dargestellt. Es wird deutlich, dass die Platzierung des Skalierungszentrums einen erheblichen Einfluss auf die Lösung hat.

Skalierungszentrum	(x, y, z)	Kommentar
SC_1	(0,0, 0,0, 0,9)	Sehr nah am Punkt C
SC_2	(0,0, 0,0, 0,0)	Im Punkt A
SC_3	(0,0, 0,0, -100,0)	Weit außerhalb des Nahfeldes

Tabelle 5.2.: Koordinaten der Skalierungszentren SC_1 bis SC_3 .

Im ersten Fall SC_1 ist das Skalierungszentrum sehr nah am Rand platziert, sodass die Querschnittsfläche schlagartig sehr stark ansteigt, wodurch der Impedanzsprung sehr groß ist und ein Großteil der Welle reflektiert wird. Das Nahfeld ist an einen beinahe unendlichen Halbraum angekoppelt. Im zweiten Fall liegt das Skalierungszentrum SC_2 im Punkt A und damit deutlich vom Rand Γ entfernt. Auch hier kommt es zu einer Reflexion der Welle am Übergang von Nah- zu Fernfeld, da auch in diesem Fall eine Querschnittsänderung vorliegt. Bei der dritten Konfiguration ist das Skalierungszentrum SC_3 sehr weit vom Rand entfernt. Es liegt weit außerhalb des Nahfeldes. Aufgrund des großen Abstandes zwischen Skalierungszentrum und Rand hat der aufgespannte Pyramidenstumpf eine kleine Steigung, wodurch der Impedanzsprung ebenfalls klein ausfällt, sodass in diesem Fall ein unendlich langer Stab approximiert wird und die Verschiebung der drei Knotenpunkte A , B und C , nachdem die Welle das Nahfeld verlassen hat, nahezu konstant bleibt. Wird das Skalierungszentrum noch weiter vom Rand entfernt, wird das berechnete Gebiet aus Nah- und Fernfeld dem Stab immer ähnlicher.

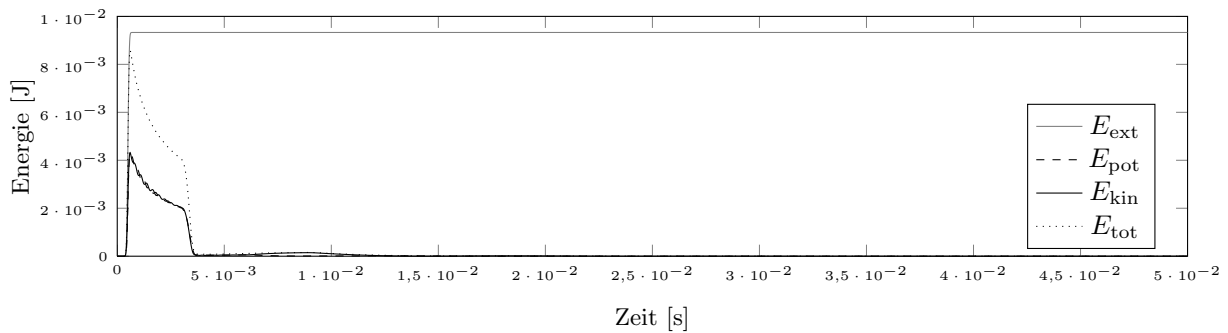
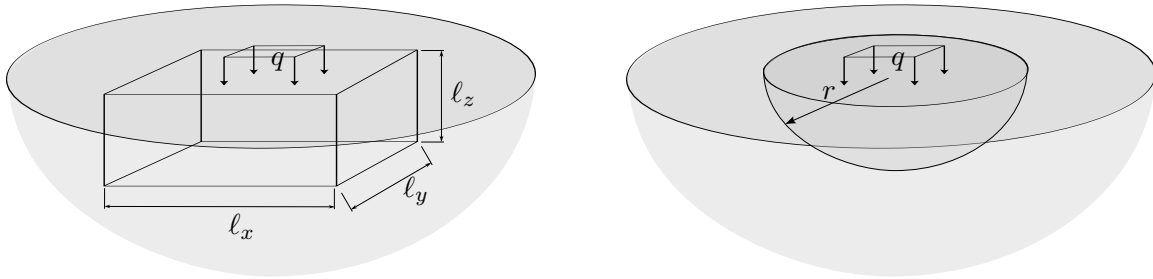


Bild 5.11.: Energieverlauf der im Bild 5.10b dargestellten Simulation.

Verlässt die Welle das Nahfeld, fällt die Energie im System sofort stark ab. Letzte Reste bleiben jedoch noch einige Zeit im Nahfeld, bis sich sämtliche durch den Impuls initiierte Spannungen wieder vollständig abgebaut haben. Bild 5.11 zeigt den zeitlichen Verlauf der Bewegungsenergien für den Fall SC_2 ; die Energieverläufe für SC_1 und SC_3 sind ähnlich.

5.2. Setzung

Die Setzung eines schlaffen Fundaments, eingebettet in einen unendlichen, isotropen und ideal-elastischen Halbraum eignet sich gut zur Demonstration des Verfahrens. Die zu untersuchen- den Geometrien sind in den Bildern 5.12a und 5.12b qualitativ dargestellt. Nah- und Fernfeld werden wie bereits eingangs diskutiert mit finiten sowie *scaled boundary* finiten Elementen diskretisiert. Ein in den Abmessungen vergleichbares, wenn auch zweidimensionales Modell, basierend auf einem gekoppelten Ansatz aus FEM und BEM haben VON ESTORFF und FIRU- ZIAAN in [159] diskutiert. In [123] beschreiben PLASSMANN et al. ein ähnliches Beispiel, das mit Hilfe von mit ANSYS gekoppelten infiniten Elementen durchgeführt wurde.



(a) Das Nahfeld ist durch einen Quader der Kantenlänge $\ell_x = \ell_y = 457,2 \text{ m}$ und $\ell_z = 190,2 \text{ m}$ beschrieben. (b) Das Nahfeld ist durch eine Halbkugel mit dem Radius $r = 190,2 \text{ m}$ beschrieben.

Bild 5.12.: Die beiden Bilder **a** und **b** zeigen die schematische Darstellung eines unendlichen Halbraums mit konstanter Flächenlast. Die Nahfeld-Fernfeld-Schnittstelle ist mit einem Quader **a** bzw. einer Halbkugel **b** diskretisiert. In beiden Beispielen wird eine Flächenlast der Größe $q = 70 \text{ kNm}^{-2}$ aufgebracht.

Die Genauigkeit der Setzungsberechnung wird im Weiteren mit Hilfe einer Konvergenzstudie im Abschnitt 5.2.2 gezeigt. Dabei wird die Auflösung der zur Berechnung verwendeten Netze sowohl im Nahfeld als auch im Fernfeld sukzessive erhöht, wodurch die Anzahl der Freiheitsgrade stetig ansteigt. Die Ergebnisse der numerischen Simulationen werden mit der nachfolgend diskutierten semi-analytischen Lösung verglichen.

5.2.1. Semi-analytischer Lösungsansatz

Die Setzung eines unendlichen, isotropen und idealelastischen Halbraumes aufgrund einer orthogonal zur Oberfläche, räumlich sowie zeitlich konstant einwirkenden Flächenlast kann mit dem von HARR vorgeschlagenen semi-analytischen Lösungsansatz erfolgen [76]. Die Lösung bei statischer Last für die vertikale Setzung an der Ecke der Flächenlast in Abhängigkeit der z -Richtung ist mit

$$s(z) = \frac{qb}{E} (1 - \nu^2) \left(A - \frac{1 - 2\nu}{1 - \nu} B \right) \quad (5.13)$$

und mit

$$A = \frac{1}{2\pi} \left(\ln \left(\frac{\sqrt{1+m^2+n^2}+m}{\sqrt{1+m^2+n^2}-m} \right) + m \ln \left(\frac{\sqrt{1+m^2+n^2}+1}{\sqrt{1+m^2+n^2}-1} \right) \right), \quad (5.14)$$

$$B = \frac{n}{2\pi} \arctan \left(\frac{m}{n\sqrt{1+m^2+n^2}} \right), \quad (5.15)$$

$$m = \frac{\ell}{b} \quad \text{und} \quad (5.16)$$

$$n = \frac{z}{b} \quad (5.17)$$

gegeben [76]. Die Parameter ℓ und b beschreiben Länge und Breite der Fläche, auf der die Last einwirkt. Für $z = 0$ ist die Lösung (5.13) nicht definiert, weshalb im Weiteren zur Ermittlung der Setzung an der Oberfläche $\lim_{z \rightarrow 0} s(z)$ ausgewertet wird. Somit gilt für $s(z = 0)$

$$s = \frac{qb}{2\pi E} (1 - \nu^2) \left(\ln \left(\frac{\sqrt{1+m^2}+m}{\sqrt{1+m^2}-m} \right) + m \ln \left(\frac{\sqrt{1+m^2}+1}{\sqrt{1+m^2}-1} \right) \right). \quad (5.18)$$

Soll die Setzung in Flächenmitte bestimmt werden, ist Gleichung (5.18) für vier Teilflächen auszuwerten und zu summieren.

5.2.2. Numerische Lösung

Die nachfolgenden Ergebnisse sind bereits in [143] ausführlich beschrieben und werden hier in einer modifizierten Darstellungsform dargestellt. Bei der hier untersuchten Setzungsberechnung wirkt auf einer Fläche von $152,4 \times 152,4 \text{ m}^2$, wie in Bild 5.12a dargestellt, eine konstante Flächenlast der Größe $q = 70 \text{ kNm}^{-2}$ ein. Das Nahfeld ist über die Kantenlängen $\ell_x = \ell_y = 457,2 \text{ m}$ und $\ell_z = 190,2 \text{ m}$ definiert. In der Mitte der einwirkenden Flächenlast liegt das Skalierungszentrum, sodass der kürzeste Abstand zwischen Skalierungszentrum und Rand Γ 190,2 m beträgt.

CSM	1	2	3	4	5
FHG _{FEM}	1500	3042	5376	9747	14520
FHG _{SBFEM}	732	1227	1848	2811	3720
$\Theta = \frac{\text{FHG}_{\text{SBFEM}}}{\text{FHG}_{\text{FEM}}}$	48,8%	40,3%	34,4%	28,8%	25,6%

Tabelle 5.3.: Quaderförmige Diskretisierung mit unterschiedlicher Anzahl an Freiheitsgraden (FHG) sowie deren Verhältniswert Θ zueinander.

Die Kurzform der Netzbezeichnung *Cuboid Shaped Mesh* (CSM) für die quaderförmige Diskretisierung ist der englischsprachigen Veröffentlichung entnommen und wird hier im Weiteren ebenfalls benutzt. Die Anzahl der Freiheitsgrade je Diskretisierung sowie der Anteil an gekoppelten Freiheitsgraden ist Tabelle 5.3 zu entnehmen. Unter der Voraussetzung, dass die Kantenlänge der Elemente in jeder Raumrichtung in etwa gleich sein soll, lässt sich die Anzahl der Freiheitsgrade im Nahfeld mit $\frac{3}{2}(n+1)^2(n+2)$ abschätzen, wobei n die Anzahl der Elemente in x-Richtung angibt. Das Größenverhältnis wird hier nur in etwa eingehalten, da $2\ell_z \neq \ell_x$ ist. Mit derselben Annahme können die Freiheitsgrade des Fernfeldes abgeschätzt werden, es gilt $3(3n^2 + 2n + 1)$. Es wird deutlich, dass die Anzahl der Freiheitsgrade im Nahfeld mit $\approx \frac{3}{2}n^3$ wächst, während die Anzahl der Freiheitsgrade im Fernfeld lediglich mit $\approx 9n^2$ zunimmt. Da jeder SBFEM Freiheitsgrad mit jeweils einem FEM Freiheitsgrad gekoppelt ist, wird das Verhältnis Θ der Kopplungsknoten in Abhängigkeit der FEM Freiheitsgrade bestimmt. Näherungsweise kann das Verhältnis von SBFEM zu FEM Freiheitsgraden vorab mit $\Theta = \frac{6}{n+3}$ bestimmt werden.

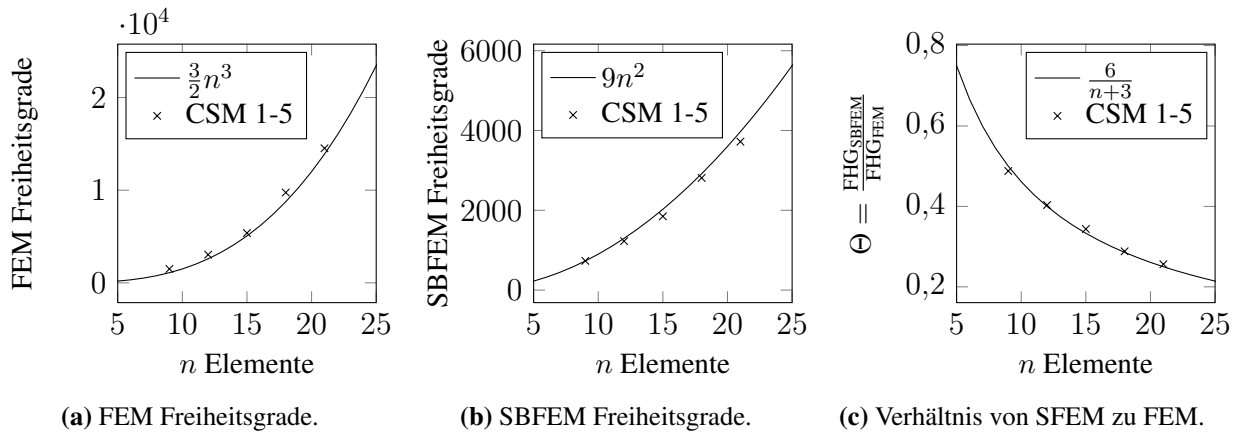
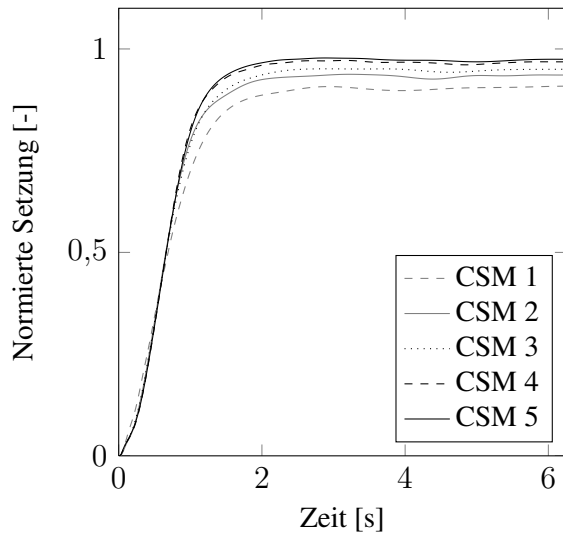


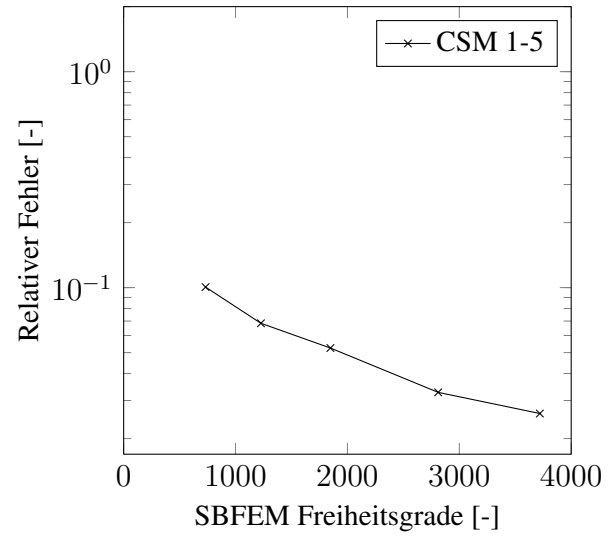
Bild 5.13.: Freiheitsgrade in Abhängigkeit von n Elementen (CSM vgl. Tabelle 5.3), **a** und **b** geben die daraus resultierende Anzahl an FEM bzw. SBFEM Freiheitsgraden wieder, **c** deren Verhältnis zueinander.

In den Bildern 5.13a bis 5.13c sind die Funktionen für die diskutierten Fälle dargestellt. Zusätzlich sind die Freiheitsgrade der Netze CMS 1 bis 5 in der jeweiligen Grafik hinzugefügt. Die mit der Netzverfeinerung einhergehende Steigerung des Rechenaufwandes wird in Kapitel 6 ausführlich diskutiert und an dieser Stelle nicht betrachtet.

Die Setzungssimulation wird für die in Tabelle 5.3 aufgeführten Netze (CSM 1 bis 6) sowie zwei unterschiedliche Materialparametersätze durchgeführt. Im ersten Fall werden die Materialparameter für den isotropen, homogenen und idealelastischen Halbraum wie folgt gewählt: $E = 37150,0 \text{ kNm}^{-2}$, $\nu = 0,48$ und $\rho = 1800,0 \text{ kgm}^{-3}$. Diese Materialparameter führen zu einer Primärwellengeschwindigkeit $c_p = 425,779 \text{ ms}^{-1}$, mit der die kritische Zeitschrittlänge $\Delta t_{\text{krit}} \leq \frac{r}{30c_p} = 0,0149 \text{ s}$ abgeschätzt werden kann [32]. Die Simulation der Setzung soll für 6,25 s erfolgen. Aus einer gewählten Zeitschrittlänge von $\Delta t = 0,0125 \text{ s}$ folgen 500 Zeitschritte. Die Zeitintegration erfolgt mit dem HHT- α Verfahren (vgl. Kapitel 3.2.2) und den in Tabelle 3.1 entsprechend ausgewiesenen Integrationsparametern.



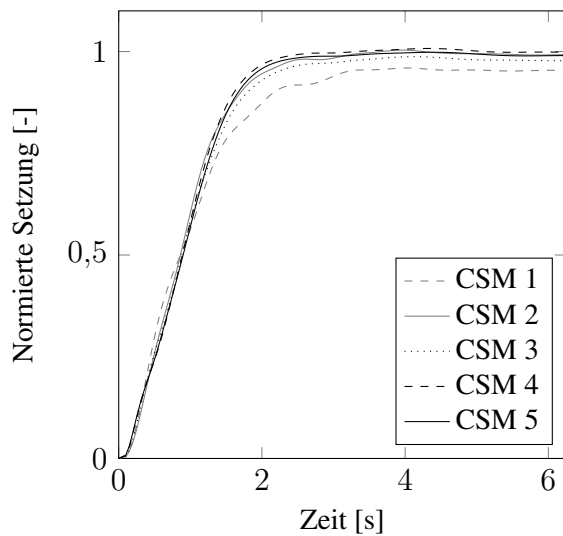
(a) Zeitveränderliche normierte Setzung.



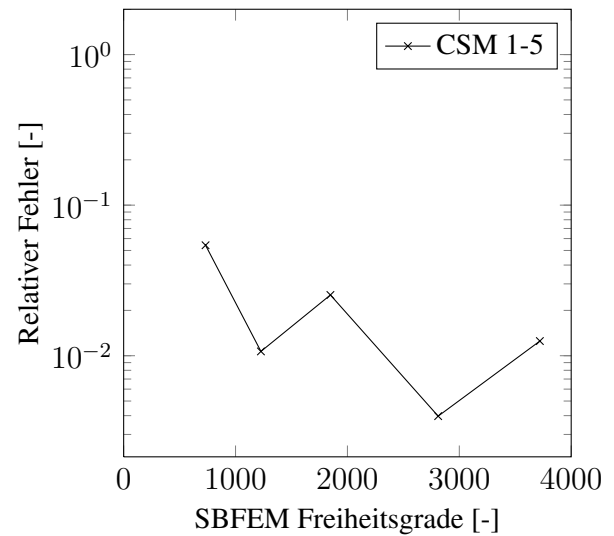
(b) Diskretisierungsabhängiger relativer Fehler.

Bild 5.14.: Ergebnisse der Setzungsberechnung unter Verwendung des CSM

mit dem Materialparametersatz: $E = 37150,0 \text{ kNm}^{-2}$, $\nu = 0,48$ und $\rho = 1800,0 \text{ kgm}^{-3}$



(a) Zeitveränderliche normierte Setzung.



(b) Diskretisierungsabhängiger relativer Fehler.

Bild 5.15.: Ergebnisse der Setzungsberechnung unter Verwendung des CSM

mit dem Materialparametersatz: $E = 21000,0 \text{ kNm}^{-2}$, $\nu = 0,13$ und $\rho = 2100,0 \text{ kgm}^{-3}$

Bild 5.14a zeigt die zeitliche Entwicklung der mit der analytischen Lösung $s = 0,248004407 \text{ m}$ normierten Setzung d . Alle fünf Netze zeigen ein sehr ähnliches Verhalten. Zu Beginn steigt die Setzung an, nach etwa zwei Sekunden erreicht die Setzung einen stationären Wert. In Bild 5.14b ist der relative Fehler

$$f = \left| \frac{s}{d} - 1 \right| \quad (5.19)$$

zum Ende der Berechnung über der Anzahl der Freiheitsgrade aufgetragen. Es wird deutlich,

dass mit Verfeinerung der Diskretisierung und somit steigender Anzahl an Freiheitsgraden die numerische Lösung gegen die semi-analytische Lösung strebt.

Im zweiten Fall werden die Materialparameter modifiziert: $E = 21000,0 \text{ kNm}^{-2}$, $\nu = 0,13$ und $\rho = 2100,0 \text{ kgm}^{-3}$. Somit folgt eine kleinere Primärwellengeschwindigkeit $c_p = 102,001 \text{ ms}^{-1}$, sodass eine größere Zeitschrittlänge von $\Delta t_{krit} \leq 0,0623 \text{ s}$ gewählt wird. Die Setzungsberechnung erfolgt für dieselbe Zeitspanne wie zuvor. Die Setzung ist in Bild 5.15a dargestellt und auf die analytische Lösung $s = 0,560443141 \text{ m}$ normiert. Der relative Fehler wird wieder in Abhängigkeit der Freiheitsgrade in Bild 5.15b dargestellt. Auch hier strebt die Lösung mit zunehmender Anzahl an Freiheitsgraden in Richtung der semi-analytischen Lösung. Für den zweiten Parametersatz ist die numerische Lösung etwa eine Größenordnung genauer im Vergleich zum vorherigen Materialparametersatz.

Analog wird die Benennung der Netze *Hemisphere Shaped Mesh* (HSM) aus [143] übernommen. Das Erzeugen halbkugelförmiger Netze ist in der Regel mit einem wesentlich höheren Aufwand verbunden, verglichen mit der Erstellung quaderförmiger Netze. HSM zeigt jedoch ein besseres Lösungsverhalten, das nachfolgend betrachtet wird. In Tabelle 5.4 sind die unterschiedlichen untersuchten Netze zusammengefasst. Der Vergleich mit Tabelle 5.3 zeigt deutlich, dass das Verhältnis von SBFEM zu FEM Freiheitsgraden mit zunehmender Netzgröße bei HSM wesentlich stärker abnimmt als beim CSM.

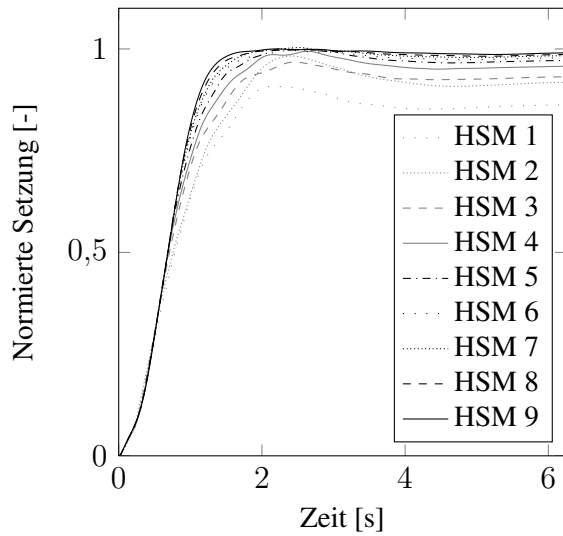
HSM	1	2	3	4	5	6	7	8	9
FHG _{FEM}	396	738	1314	3096	6030	9033	14655	19818	49155
FHG _{SBFEM}	123	219	291	480	843	1083	1515	1827	3603
$\Theta = \frac{\text{FHG}_{\text{SBFEM}}}{\text{FHG}_{\text{FEM}}}$	31,1%	29,7%	22,2%	15,5%	14%	12%	10,3%	9,2%	7,3%

Tabelle 5.4.: Halbkugelförmige Diskretisierung mit unterschiedlicher Anzahl an Freiheitsgraden (FHG) sowie deren Verhältnis zueinander.

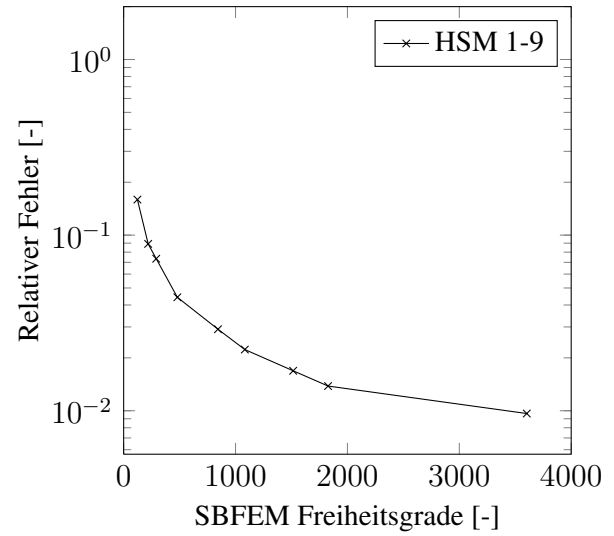
Auf die Halbkugel wirkt, wie in Bild 5.12b dargestellt, auf einer Fläche von $152,4 \times 152,4 \text{ m}^2$ ebenfalls eine konstante Flächenlast $q = 70 \text{ kNm}^{-2}$ ein. Der Abstand vom Mittelpunkt der Flächenlast, in dem sich das Skalierungszentrum der SBFEM befindet, zum Rand Γ ist gegeben mit dem Radius $r = 190,5 \text{ m}$. Somit entspricht der Abstand zwischen Skalierungszentrum und Rand dem Abstand im vorherigen Beispiel. Selbiges gilt für Material- sowie Zeitintegrationsparameter.

Die Bilder 5.16a und 5.16b zeigen die Ergebnisse des ersten Parametersatzes. Auch hier wird deutlich, dass die Genauigkeit mit zunehmender Netzauflösung zunimmt und somit der relative Fehler gleichzeitig abnimmt.

In den Bildern 5.17a und 5.17b sind die Ergebnisse des zweiten Parametersatzes zusammengefasst, die den Ergebnissen der vorangegangenen Berechnung sehr ähnlich sind. Die Darstellung des relativen Fehlers macht den Vorteil der HSM gegenüber der CSM deutlich erkennbar. Wie zu erwarten, verbessert die höhere Auflösung im Nahfeld die Genauigkeit der Simulation und

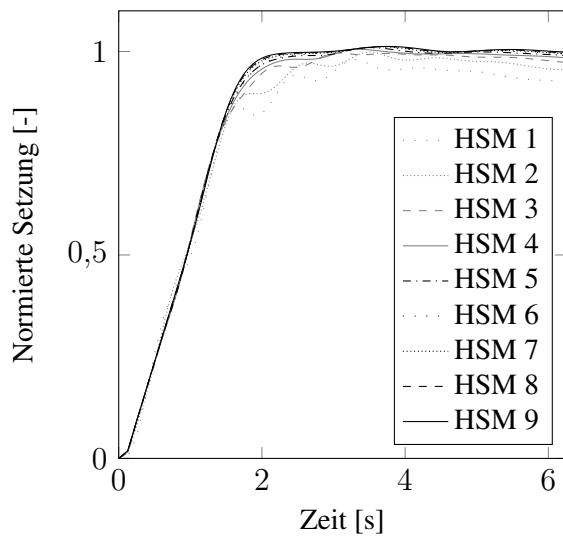


(a) Zeitabhängige normierte Setzung.

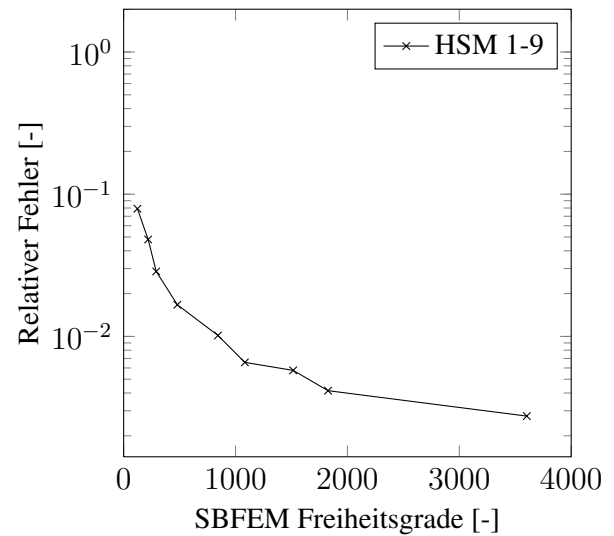


(b) Diskretisierungsabhängiger relativer Fehler.

Bild 5.16.: Ergebnisse der Setzungsberechnung unter Verwendung des HSM
mit dem Materialparametersatz: $E = 37150,0 \text{ kNm}^{-2}$, $\nu = 0,48$ und $\rho = 1800,0 \text{ kgm}^{-3}$



(a) Zeitabhängige normierte Setzung.



(b) Diskretisierungsabhängiger relativer Fehler.

Bild 5.17.: Ergebnisse der Setzungsberechnung unter Verwendung des HSM
mit dem Materialparametersatz: $E = 21000,0 \text{ kNm}^{-2}$, $\nu = 0,13$ und $\rho = 2100,0 \text{ kgm}^{-3}$

damit das Ergebnis erheblich. Der relative Fehler fällt bei dem HSM bei etwa gleicher Anzahl an Kopplungsknoten eine Größenordnung kleiner aus. Zusätzlich macht sich das Verhältnis von FEM Freiheitsgraden zu SBFEM Freiheitsgraden bei den HSM positiv bemerkbar. Bei einer hohen Auflösung im Nahfeld werden vergleichsweise wenige Kopplungsknoten am Gebietsrand benötigt, wodurch der rechenintensive Teil der Kopplung merklich reduziert wird. Wie zeitintensiv die Bestimmung des Fernfeldes und die Kopplung von Nah- und Fernfeld ist, wird, wie bereits erwähnt, im nachfolgenden Kapitel 6 erläutert.

6. Effizienzsteigerung des gekoppelten Ansatzes

Ein Nachteil der FEM-SBFEM Kopplung im Zeitbereich ist zweifellos der hohe Speicheraufwand, der sich aufgrund der vorzuhaltenden M^∞ Matrizen einstellt. Für eine Simulation von n Zeitschritten müssen in der ursprünglichen Formulierung die Matrizen eines Interfaces mit N Freiheitsgraden gespeichert werden. Da diese Einflussmatrizen in der Regel vollbesetzt sind, sind $N \times N \times n$ Werte zu speichern. Hinzu kommen weitere $N \times n$ Werte für die Knotengeschwindigkeiten \dot{u} am Interface. Ein weiterer Nachteil folgt aus dem zu bestimmenden Faltungsintegral (vgl. Gleichung (3.73) und (3.76)). Der Berechnungsaufwand steigt mit der Anzahl der zu simulierenden Zeitschritte, da für jeden Zeitschritt t_n insgesamt $n - 1$ Matrix-Vektor-Multiplikationen auszuwerten sind. Diese beiden genannten Nachteile lassen sich nicht vollständig beheben. Es ist jedoch möglich, sowohl Speicher- als auch Rechenaufwand signifikant zu reduzieren [95, 128, 168]. Nachfolgend werden unterschiedliche Methoden der Modellreduktion erläutert. Der Einfluss der hier vorgestellten Effizienzsteigerung wird anhand einer Setzungssimulation (vgl. Abschnitt 5.2) untersucht, aufgezeigt und bewertet. Die systematische Untersuchung des Einflusses der unterschiedlichen Ansätze und zusätzlich deren Kombination zeigt das große Potential, das in der Modellreduktion des Fernfeldes steckt sowie die lineare Unabhängigkeit der einzelnen Verfahren zueinander.

6.1. Extrapolation der Einflussmatrizen

Die Werte der Einflussmatrizen weisen ab einem Zeitschritt $t_n > t_m$ ein lineares Verhalten auf, die Einträge in den Matrizen wachsen in jedem Zeitschritt um einen konstanten Wert an. Dies zeigt die Auswertung der Matrixeinträge. Somit können die Werte der Einflussmatrizen für $t_n > t_m$ extrapoliert werden. Auf Grundlage dieser Beobachtung stellt LEHMANN 2003 für zweidimensionale Simulationen einen effizienten Ansatz zum Lösen des Faltungsintegrals (vgl. Gleichung (3.76)) vor [95]. Der Ansatz nutzt einen rekursiven Algorithmus, mit dessen Hilfe die Einflussmatrizen extrapoliert werden können. 2004 zeigen LEHMANN et al., dass der Ansatz auch auf dreidimensionale Aufgabenstellungen anwendbar ist [99]. Als Referenzwert zur Bestimmung, ab wann die Matrixeinträge um einen konstanten Wert anwachsen, kann wie bei LEHMANN ein Eintrag auf der Hauptdiagonalen der Matrizen dienen. Es ist jedoch auch möglich, die zeitliche Veränderung der Gesamtnorm

$$\|M^\infty\|_G = N \max_{i,j=1,\dots,N} |m_{ij}| \quad (6.1)$$

oder auch der Frobeniusnorm

$$\|M^\infty\|_F = \sqrt{\sum_{i=1}^N \sum_{j=1}^N m_{ij}^2} \quad (6.2)$$

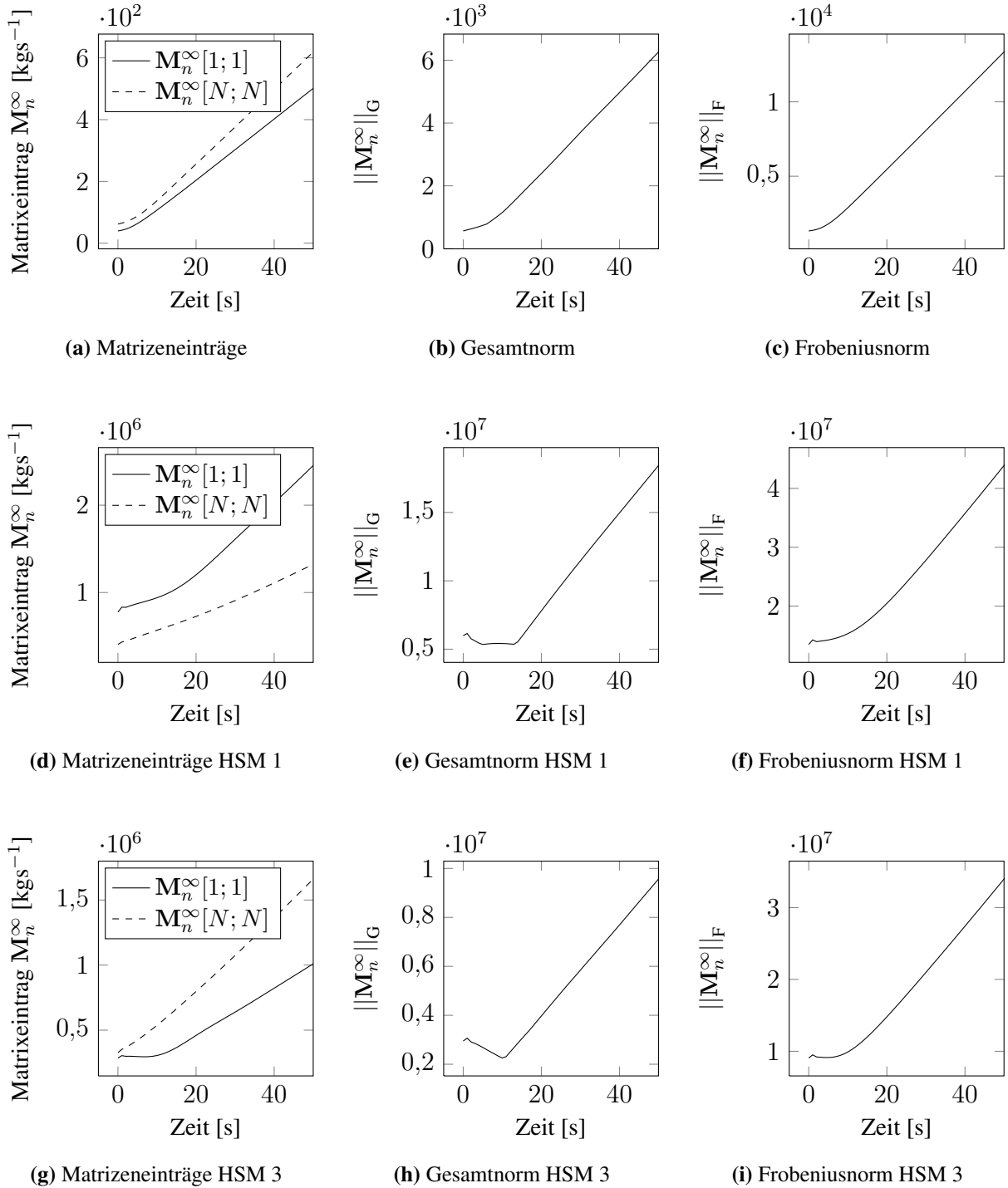


Bild 6.1.: Zeitliche Entwicklung der Einflussmatrizen. Ausgewertet und dargestellt sind die Matriceinträge $\mathbf{M}_n^\infty[1;1]$ und $\mathbf{M}_n^\infty[N;N]$, die Gesamtnorm $\|\mathbf{M}_n^\infty\|_G$ sowie die Frobeniusnorm $\|\mathbf{M}_n^\infty\|_F$ der Einflussmatrizen aus Abschnitt 5.1 (a, b und c), HSM 1 (d, e und f) und HSM 3 (g, h und i) aus Abschnitt 5.2.2. Obwohl die Auswertung der Matrizen bei allen drei Netzen auf die gleiche Art und Weise erfolgt, sind deutliche Unterschiede im zeitlichen Verlauf auszumachen.

der Einflussmatrizen als Kriterium heranzuziehen. Inwieweit sich die Wahl der Referenzgröße (Matrizeintrag $M_n^\infty[i,j]$, Gesamtnorm $\|M_n^\infty\|_G$ oder Frobeniusnorm $\|M_n^\infty\|_F$) auf die Extrapolation auswirkt, ist offen und wird nachfolgend systematisch untersucht.

Die Bilder 6.1a bis 6.1i zeigen exemplarisch die zeitliche Veränderung der Matrixwerte beziehungsweise deren Norm innerhalb der ersten 50 Zeitschritte. Die Datengrundlage der Bilder 6.1a bis 6.1c bildet das Beispiel aus Abschnitt 5.1, für Bilder 6.1d bis 6.1i basieren die Daten auf den Einflussmatrizen der Netze HSM 1 sowie HSM 3 aus Abschnitt 5.2.2. Aus den Diagrammen lassen sich verschiedene Schlüsse ziehen:

Erstens verhalten sich die Einträge in den Matrizen nicht in derselben Weise. Dies ist deutlich an den hier dargestellten Funktionsverläufen für $M_n^\infty[1;1]$ und $M_n^\infty[N;N]$ zu sehen. Daraus folgt, dass der als Referenzwert ausgewählte Matrizeintrag einen Einfluss auf den Zeitpunkt t_m hat, von dem an die Extrapolation der Matrizen erfolgt. Dies bedeutet gleichzeitig, dass der ausgewählte Referenzwert die Lösung beeinflusst. Demnach ist der bisherige Ansatz, einen beliebigen Wert auf der Hauptdiagonalen zur Bestimmung der Matrixeigenschaften heranzuziehen, nicht zufriedenstellend.

Zweitens liefert die Gesamtnorm brauchbare, aber numerisch ungünstige Werte, da die Algorithmen zur Erkennung des konstanten Wachstums der Matrizeinträge gegebenenfalls den Zeitpunkt t_m an einer unzulässigen Stelle erkennen (vgl. Bild 6.1e oder 6.1h), dies führt zwangsläufig zu falschen Lösungen.

Drittens scheint die Frobeniusnorm die aussagekräftigsten Werte zu liefern. Was zweifelsohne daran liegt, dass dieser Ansatz als einziger alle Matrizeinträge berücksichtigt. Allerdings ist dieser Ansatz von den hier vorgestellten der aufwendigste.

Es bleibt festzuhalten, dass die Entwicklung der Matrizeneinträge problemabhängig ist und alle drei Methoden zum Nachweis des linearen Verhaltens herangezogen werden können. Es empfiehlt sich jedoch, mit dem Prüfen auf Linearität der Matrizen erst zu beginnen, wenn eine gewisse Anzahl an Zeitschritten bereits berechnet wurde. Auch hier sei darauf hingewiesen, dass die Anzahl an Zeitschritten sowohl von der Geometrie des Problems als auch von der Zeitschrittlänge der Simulation abhängig ist.

Der hier beschriebene Lösungsansatz beruht auf den bereits erwähnten Veröffentlichungen von LEHMANN. Bei der Extrapolation werden die Einflussmatrizen für jeden Zeitschritt t_n mit $n > m$ in zwei Matrizen aufgeteilt, so dass

$$M_n^\infty = T^\infty t_n + C^\infty \quad (6.3)$$

gilt. C^∞ ist eine konstante Matrix, T^∞ beschreibt den Gradienten der Einflussmatrix $\left(\frac{\Delta M^\infty}{\Delta t}\right)$. Für äquidistante Zeitschritte ist der Gradient durch

$$T^\infty = M_{m+1}^\infty - M_m^\infty \quad (6.4)$$

definiert und somit ebenfalls konstant.

Die Summe aus Gleichung (3.76) wird in zwei Teilsummen zerlegt. Die erste Teilsumme bildet die Summe unter Berücksichtigung der Einflussmatrizen für die Zeitschritte $t_n, 1 \leq n \leq m$, die zweite Teilsumme bildet die Summe unter Berücksichtigung der extrapolierten Einflussmatrizen für die Zeitschritte $t_n, n > m$, somit gilt:

$$\begin{aligned} \mathbf{p}_b(t_n) &= \gamma \Delta t \mathbf{M}_0^\infty \ddot{\mathbf{u}}_n + \mathbf{p}_b^l(t_n) + \mathbf{p}_b^{\text{nl}}(t_n), \\ &= \gamma \Delta t \mathbf{M}_0^\infty \ddot{\mathbf{u}}_n + \sum_{j=1}^{n-m+1} \mathbf{M}_{n-j+1}^\infty (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}) + \sum_{j=n-m+2}^{n-1} \mathbf{M}_{n-j+1}^\infty (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}). \end{aligned} \quad (6.5)$$

Das Lösen des Terms $\mathbf{p}_b^{\text{nl}}(t_n)$ erfolgt analog zu Gleichung (3.76). Der Term $\mathbf{p}_b^l(t_n)$ wird in einen rekursiven Algorithmus überführt. Einsetzen der Gleichung (6.3) in den linearen Term aus Gleichung (6.5) liefert

$$\begin{aligned} \mathbf{p}_b^l(t_n) &= \sum_{j=1}^{n-m+1} (\mathbf{T}^\infty t_{n-j+1} + \mathbf{C}^\infty) (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}), \\ &= (\mathbf{T}^\infty t_m + \mathbf{C}^\infty) (\dot{\mathbf{u}}_{n-m+1} - \dot{\mathbf{u}}_{n-m}) + \sum_{j=1}^{n-m} (\mathbf{T}^\infty t_{n-j+1} + \mathbf{C}^\infty) (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}), \quad (6.6) \\ &= \mathbf{M}_m^\infty (\dot{\mathbf{u}}_{n-m+1} - \dot{\mathbf{u}}_{n-m}) + \sum_{j=1}^{n-m} (\mathbf{T}^\infty t_{n-j+1} + \mathbf{C}^\infty) (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}). \end{aligned}$$

Für den rekursiven Algorithmus wird $\mathbf{p}_b^l(t_{n-1})$ benötigt, somit ist n durch $n - 1$ in Gleichung (6.6) zu ersetzen

$$\mathbf{p}_b^l(t_{n-1}) = \sum_{j=1}^{n-m} (\mathbf{T}^\infty t_{n-j} + \mathbf{C}^\infty) (\dot{\mathbf{u}}_j - \dot{\mathbf{u}}_{j-1}) \quad (6.7)$$

und die Differenz aus $\mathbf{p}_b^l(t_n)$ und $\mathbf{p}_b^l(t_{n-1})$ zu bilden. Daraus folgt letztlich die Rekursionsvorschrift

$$\mathbf{p}_b^l(t_n) = \mathbf{p}_b^l(t_{n-1}) + \mathbf{M}_m^\infty (\dot{\mathbf{u}}_{n-m+1} - \dot{\mathbf{u}}_{n-m}) + \mathbf{T}^\infty (\dot{\mathbf{u}}_{n-m} - \dot{\mathbf{u}}_0). \quad (6.8)$$

Für jeden weiteren Zeitschritt ist analog zu verfahren. Auf Grundlage dieser Modellreduktion sind anstelle der $N \times N \times n$ Werte der Einflussmatrizen nur $N \times N \times m$ Werte zu speichern. Die zuvor erwähnten Knotengeschwindigkeiten am Interface müssen trotz der Optimierung der Einflussmatrizen für jeden Zeitschritt gespeichert werden, somit sind weiterhin $N \times n$ Werte für die Knotengeschwindigkeiten am Interface bereit zu stellen. Der Einfluss der Linearisierung auf die Lösung wird nachfolgend diskutiert.

Die Rechenzeiterparnis wird besonders groß bei Langzeitsimulationen, bei denen das Verhältnis $\psi = \frac{m}{n}$ möglichst klein ist. LEHMANN hat 2005 gezeigt, dass mit Hilfe dieses Verfahrens der Berechnungsaufwand fallabhängig um bis zu 90% reduziert werden kann [99].

Einen anderen sehr effektiven Algorithmus zur Reduktion der Rechenzeit stellten RADMANOVIĆ und KATZ 2010 vor [128]. Auch sie nutzen die konstanten Zuwachsraten der Einflussmatrizen aus. Zusätzlich bedienen sie sich zweier unterschiedlicher Zeitskalen für Nahfeld und Fernfeld, so dass sie die Anzahl der Einflussmatrizen ebenfalls reduzieren können. Bei der Berechnung der Einflussmatrizen nutzen sie größere Zeitschritte als bei der Simulation des gekoppelten Nah- und Fernfeldes. Fehlende Fernfeldinformationen werden aus den vorhandenen Daten interpoliert.

Bisher wird in den bereits genannten Veröffentlichungen lediglich gezeigt, dass die Extrapolation der Matrizen möglich ist und ähnliche bis deckungsgleiche Kurven für die Lösungen berechnet werden können. Welchen Einfluss die Wahl des Zeitschrittpunkts t_m hat, bleibt jedoch offen. Nachfolgend wird für ausgewählte Zeitschritte t_m der benötigte Speicherbedarf sowie die Rechenzeit ermittelt und mit dem relativen Fehler der so bestimmten Lösung verglichen.

Die Untersuchung des Einflusses der Extrapolation der Einflussmatrizen auf die Lösung erfolgt anhand der zuvor diskutierten Setzungsberechnung. Die Setzungssimulation wird mit dem CSM 5 sowie dem HSM 9 durchgeführt, unter der Annahme, dass die Einflussmatrizen M^∞ nach einem definierten Zeitpunkt t_m extrapoliert werden können. Materialparameter ($E = 21000,0 \text{ kNm}^{-2}$, $\nu = 0,13$ und $\rho = 2100,0 \text{ kgm}^{-3}$) und Zeitintegrationsparameter entsprechen denen aus dem Abschnitt 5.2.2. Im Falle der Simulation werden lediglich m Einflussmatrizen berechnet, alle übrigen Matrizen werden extrapoliert.

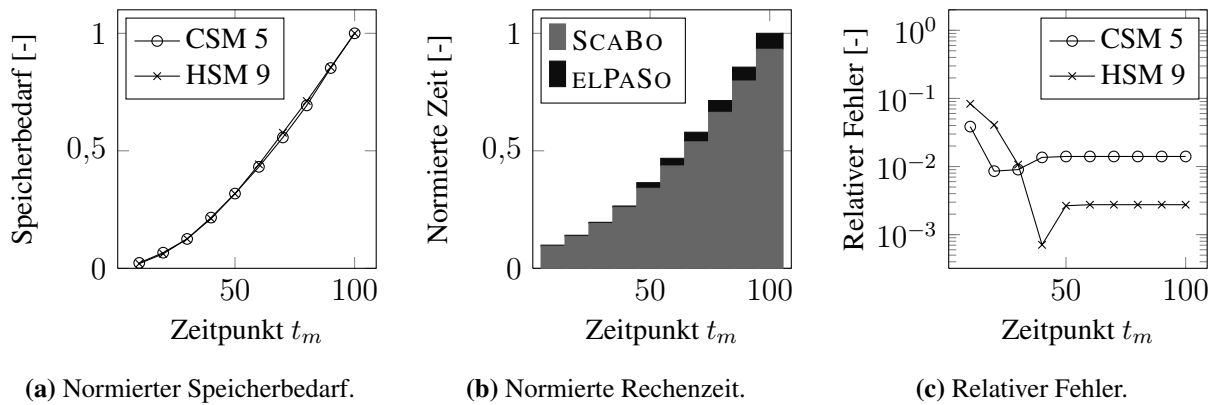


Bild 6.2.: Normierter Speicherbedarf der Einflussmatrizen M^∞ (a), normierte Rechenzeit (b) sowie der relative Fehler der Setzungsberechnung (c) in Abhängigkeit der Modellreduktion für das Netz HSM 9.

Der benötigte normierte Speicherbedarf sowie die normierte Rechenzeit sind in Bild 6.2a beziehungsweise 6.2b dargestellt. Dargestellt sind zwei normierte Rechenzeiten, zum einen die zur Bestimmung der Einflussmatrizen des Fernfeldes (SCABO) und zum anderen die zur Setzungsberechnung des gekoppelten Systems bestehend aus Nah- und Fernfeld (ELPASO). Dabei sind sämtliche Rechenzeiten im Verhältnis zur ursprünglichen Berechnung ($m = n = 100$) aus Abschnitt 5.2.2 angegeben. Auf die absolute Simulationszeit wird hier nicht eingegangen, da sie vom Computer und der Anzahl der verwendeten Prozesse abhängig ist, dies wird in Kapitel 4.1

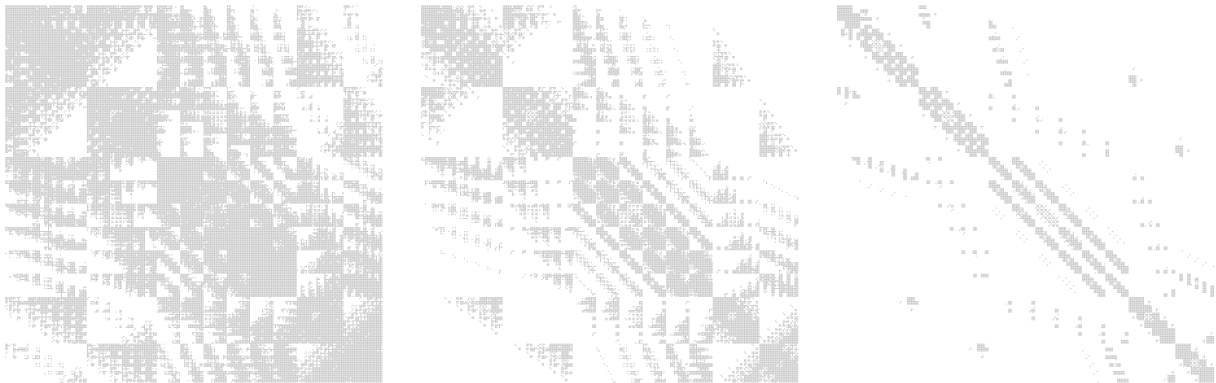
thematisiert und genauer untersucht. Es ist leicht ersichtlich, dass mit zunehmender Anzahl an Einflussmatrizen der Speicherbedarf zunimmt. Gleichzeitig steigt ebenfalls der Rechenaufwand und damit zwangsläufig die Simulationszeit.

Der relative Fehler (vgl. Gleichung (5.19)) der Setzungsberechnung ist in Bild 6.2c dargestellt. Simulationen mit weniger als 50 Einflussmatrizen konvergieren nicht zu der gewünschten Lösung. Der vermeintlich kleinste Fehler tritt bei $t_m = 40$ auf, wobei die Lösung nicht stabil ist und somit als Ausreißer zu bewerten ist.

Es bleibt festzuhalten, dass mit der Methode bei gleichbleibendem relativen Fehler der Rechenaufwand auf circa 37% des ursprünglichen Aufwands und der Speicherbedarf auf circa 32% der Ausgangsgröße reduziert werden kann. Wird t_m zu klein gewählt und somit zu früh mit der Extrapolation der Matrixwerte begonnen, beeinflusst dies die Lösung.

6.2. Geometrische Entkopplung

Die Einflussmatrizen sind in der Regel voll besetzt, was bedeutet, dass alle Knoten auf der Nahfeld-Fernfeld-Schnittstelle miteinander gekoppelt sind. Je nach geometrischem Abstand der Knoten zueinander fällt diese Kopplung unterschiedlich stark aus. Die Position der Freiheitsgrade innerhalb der Matrix ist dabei nicht ausschlaggebend. Knoten mit einem geringen geometrischen Abstand haben einen größeren Einfluss aufeinander als Knoten mit einem großen Abstand. Wie groß der Einfluss zweier Knoten aufeinander ist, bestimmen die Werte der Einflussmatrix. Je geringer der Einfluss zweier Knoten aufeinander ausfällt, desto kleiner sind die Werte.



(a) Belegung der Einflussmatrix $\epsilon_z = 1,0 \cdot 10^{-7}$. (b) Belegung der Einflussmatrix $\epsilon_z = 1,0 \cdot 10^{-5}$. (c) Belegung der Einflussmatrix $\epsilon_z = 1,0 \cdot 10^{-3}$.

Bild 6.3.: Belegung der Einflussmatrix M_0^∞ auf Grundlage der Netze HSM 3: Dargestellt sind Matrixeinträge größer der gewählten Toleranz ϵ_z .

Bei der geometrischen Entkopplung wird eine untere Schranke ϵ_z definiert. Die untere Schranke bestimmt wie groß ein Wert in der Matrix mindestens sein muss, damit die Kopplung der Knoten

berücksichtigt werden soll, beziehungsweise wann der gegenseitige Einfluss so klein ist, dass er vernachlässigt werden kann.

Die Bilder 6.3a bis 6.3c zeigen die Belegung der Einflussmatrix M_0^∞ für unterschiedliche untere Schranken ϵ_z , im Bereich von $1,0 \cdot 10^{-7}$ bis $1,0 \cdot 10^{-3}$. Dabei ist jeder Wert, der ungleich Null ist, mit einem Punkt dargestellt. Die Datengrundlage bilden die Matrizen des Beispiels HSM 3 aus Abschnitt 5.2.2. Die dargestellten Matrizen haben $291 \times 291 = 84681$ Einträge, wenn keine untere Schranke definiert ist. Wird $\epsilon_z = 1,0 \cdot 10^{-7}$ gewählt, resultieren 49855 Nicht-Nullwerte, was einer Belegung von 58,87% entspricht (vgl. Bild 6.3a). Wird ϵ_z auf $1,0 \cdot 10^{-5}$ angehoben, ist die Matrix, wie in Bild 6.3b dargestellt, nur noch zu 28,67% belegt. Bei einer unteren Schranke von $\epsilon_z = 1,0 \cdot 10^{-3}$ fällt die Anzahl der Nicht-Nullwerte auf 7203 bzw. 8,51% verglichen mit der Anzahl der vollbesetzten Matrix. Reduktionen ähnlicher Größenordnungen werden bereits von BORSUTZKY und LEHMANN vorgestellt (vgl. [32, 98]).

Die geometrische Entkopplung reduziert lediglich die Menge der zu speichernden Daten, die Rechenzeit wird nicht beeinflusst, da das vollständige Fernfeld diskretisiert und berechnet wird. Es ist jedoch möglich, die zuvor erläuterte Extrapolation der Einflussmatrizen mit der geometrischen Entkopplung zu kombinieren um Rechenzeit und Speicher einzusparen.

Nachfolgend wird der Einfluss der geometrischen Entkopplung am Netz HSM 9 untersucht. Dafür werden Einflussmatrizen mit unterschiedlich gewählten geometrischen Toleranzen ϵ_z ausgewertet. Die Toleranzen liegen im Bereich von $\epsilon_z = 1,0 \cdot 10^{-8}$ bis $\epsilon_z = 1,0 \cdot 10^{-2}$. Die Beurteilung der Modellreduktion erfolgt hinsichtlich des Speicherbedarfs, der Rechenzeit sowie des relativen Fehlers.

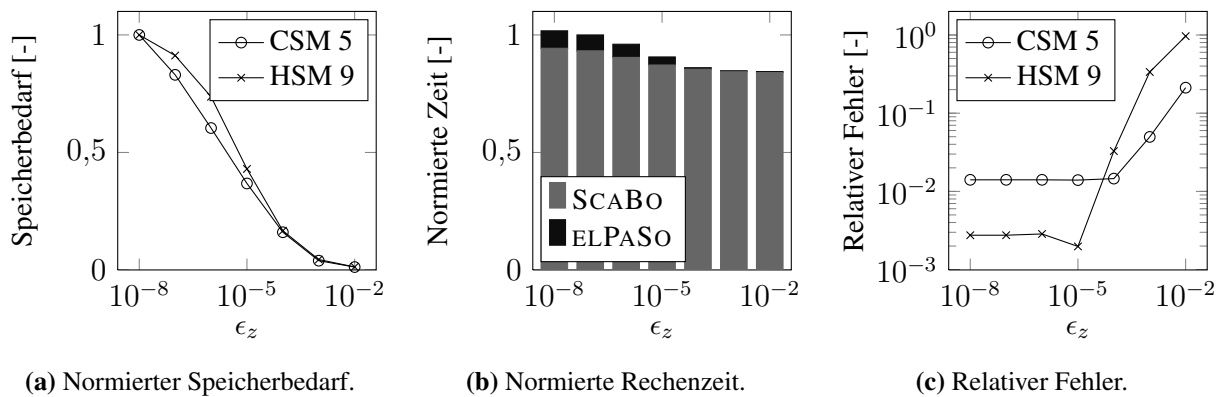


Bild 6.4.: Normierter Speicherbedarf der Einflussmatrizen M^∞ (a), normierte Rechenzeit (b) sowie relativer Fehler der Setzungsberechnung (c) in Abhängigkeit der geometrischen Entkopplung mit der Toleranz ϵ_z für das Netz HSM 9.

Bei allen vorangegangenen Simulationen werden Werte kleiner $\epsilon_z = 1,0 \cdot 10^{-7}$ in der Einflussmatrix nicht berücksichtigt. Die Simulationen aus Abschnitt 5.2.2 werden hier als Referenzlösung hinsichtlich des Speicherbedarfs und der Rechenzeit herangezogen. Die Bilder 6.4a bis 6.4c zeigen wie schon zuvor den Speicherbedarf der Einflussmatrizen, die Rechenzeit sowie den relativen Fehler der Setzungsberechnung, jetzt jedoch in Abhängigkeit der Toleranz ϵ_z .

Es ist wesentlich, dass die Wahl der Toleranz einen signifikanten Einfluss auf die Größe der zu speichernden Matrizen hat. Je größer ϵ_z gewählt wird, desto geringer fällt der Speicherbedarf aus. Gleichzeitig steigt allerdings auch der relative Fehler der Simulation. Im Vergleich zur Modellreduktion durch Extrapolation der Einflussmatrizen ist die Rechenzeit in diesem Fall nahezu konstant. Auch bei großen Toleranzen $\epsilon_z \geq 1,0 \cdot 10^{-4}$ liegt die Rechenzeit bei circa 85% der Vergleichssimulation. Der Grund dafür ist offensichtlich: Bei konstanter Systemgröße bleibt auch die Rechenzeit konstant. Die Zeitersparnis lässt sich also auf den Lese- und Schreibprozess der Einflussmatrizen zurückführen.

Extrapolation und geometrische Entkopplung können wie bereits erwähnt kombiniert werden. Wird beispielsweise $t_m = 50$ und $\epsilon_z = 1,0 \cdot 10^{-5}$ gewählt, dann benötigt die Simulation, verglichen mit der Referenzlösung ($t_m = 100$, $\epsilon_z = 1,0 \cdot 10^{-7}$), 33% der ursprünglichen Rechenzeit und 17,3% des Speicherbedarfs der Einflussmatrizen bei gleichem relativen Fehler. Dabei entspricht die neue Rechenzeit dem Produkt der relativen Rechenzeitreduzierungen der einzelnen Modellreduktionen ($t_m = 50$, $\epsilon_z = 1,0 \cdot 10^{-5}$ mit 36,4% bzw. 90,6% der ursprünglichen Rechenzeiten vgl. Bild 6.2b und 6.4b).

6.3. Zerlegung des Fernfeldes in Substrukturen

Wie bereits erwähnt, wird die Rechenzeit durch die geometrische Entkopplung nicht beeinflusst, wenn die Entkopplung nach der Berechnung der vollständigen Einflussmatrizen erfolgt. Alternativ ist es möglich, die Entkopplung im Vorfeld zu realisieren, indem das Fernfeld in Substrukturen aufgeteilt wird [161]. Dabei werden voneinander weit entfernte Knoten, die einen geringen Einfluss aufeinander haben, durch Einführen künstlicher Ränder voneinander getrennt. Jede einzelne Substruktur beinhaltet nur noch einen Teil des vollständigen Fernfeldes und kann unabhängig von allen anderen Substrukturen berechnet werden. Die Bilder 6.5a bis 6.5c zeigen ein in zwei bis vier Substrukturen aufgeteiltes Fernfeld.

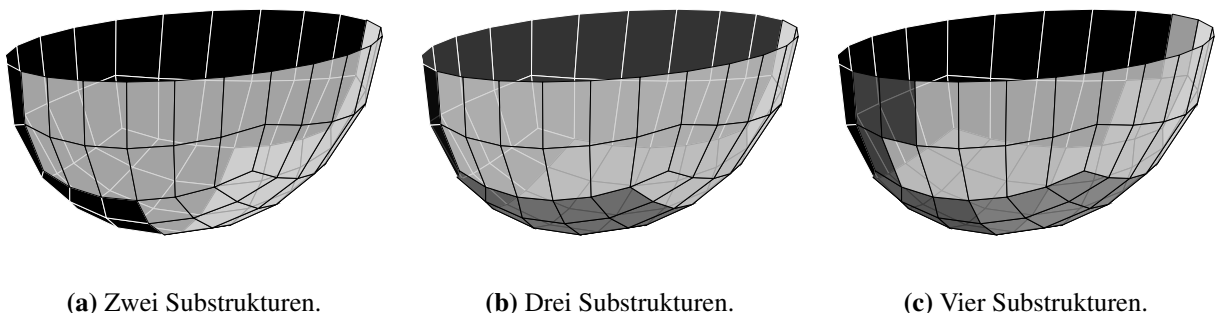


Bild 6.5.: Zerlegtes Fernfeld in zwei (a), drei (b) und vier (c) Substrukturen am Beispiel des HSM 3.

Die künstlichen Ränder bewirken jedoch auch eine Trennung direkter Nachbarknoten mit vergleichsweise großem gegenseitigen Einfluss. In den Anwendungen zeigt sich jedoch, dass der

Ansatz brauchbare Ergebnisse liefert, wenn die Aufteilung in Teilstrukturen die Topologie des Elementnetzes berücksichtigt. Dies ist auf die Kopplung von FEM und SBFEM zurückzuführen, da bei der elementweisen Zerlegung des Fernfeldes Überlappungen der Einflussmatrizen aufgrund gemeinsamer Knoten bei der gekoppelten Berechnung resultieren. Die Kopplung der einzelnen Substrukturmatrizen ist in den Bildern 6.6a bis 6.6c dargestellt. Die Struktur der vollständigen Matrix (vgl. Bild 6.3a) ist weiterhin erkennbar. Die Belegung der Matrizen M_0^∞ jeder Substruktur ist in Tabelle 6.1 zusammengefasst. Dabei wird deutlich, dass mit steigender Anzahl an Substrukturen die Matrixgröße abnimmt und der Anteil der Nicht-Nullwerte stetig zunimmt. Je kleiner die Substruktur, desto größer ist der Anteil an Nicht-Nullwerten. Dies bedeutet, dass die Knoten innerhalb der Substrukturen eine relativ starke Kopplung aufweisen.

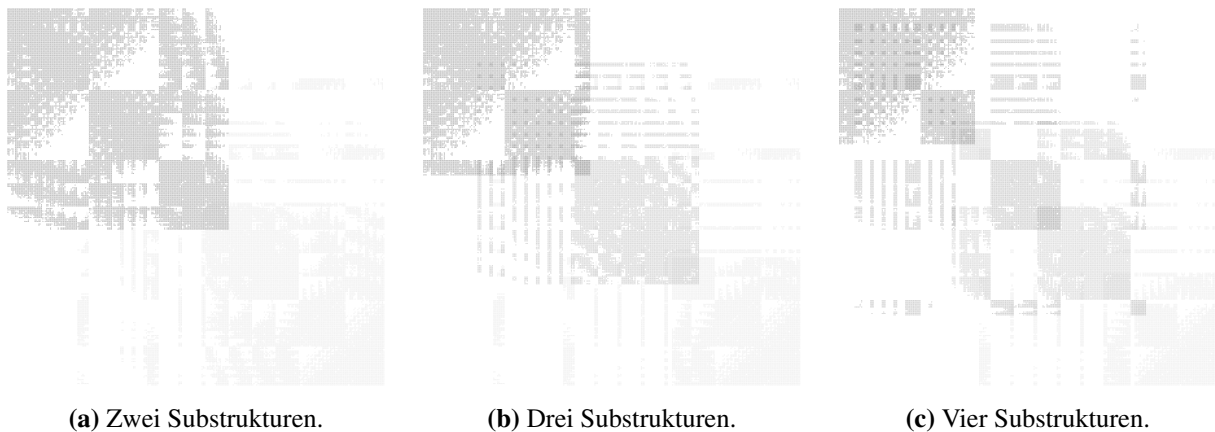


Bild 6.6.: Belegung der Einflussmatrix M_0^∞ des in zwei (a), drei (b) und vier (c) Substrukturen zerlegten Fernfelds: Matrixeinträge größer $\epsilon_z = 1,0 \cdot 10^{-7}$ am Beispiel des HSM 3.

Tabelle 6.1.: Größe und Belegung der Substrukturmatrizen M_0^∞ .

Substrukturen		Matrixgröße	Einträge	Nicht-Null Werte	Anteil %
2	A	171×171	29241	19601	67,03
	B	171×171	29241	19605	67,05
3	A	129×129	16641	11665	70,10
	B	123×123	15129	11567	76,46
	C	129×129	16641	11673	70,15
4	A	105×105	11025	8105	73,51
	B	102×102	10404	8102	77,87
	C	99×99	9801	8015	81,78
	D	105×105	11025	8125	73,70

Wird die Gebietszerlegung mit der geometrischen Entkopplung kombiniert und ϵ_z groß genug gewählt, dann entsprechen die zusammengesetzten Einflussmatrizen der einzelnen Substrukturen den Einflussmatrizen des gesamten Fernfeldes nahezu vollständig (vgl. [32]). Für kleine ϵ_z ist dies nicht der Fall.

Theoretisch ist auch eine Aufteilung der Substrukturen anhand der Knoten möglich. Dies würde die Datenmenge weiter reduzieren, da die Knoten der Schnittstelle nur noch in einer der Substrukturen auftauchen würden. Dies bedeutet allerdings auch, dass ein Teil der Elemente verloren ginge und der Einfluss des unendlichen Halbraumes nicht vollständig erfasst werden würde. Überlappungen der Einflussmatrizen der Substrukturen sind ausgeschlossen, da es keinerlei gemeinsame Knoten gibt. Sollten die fehlenden Elemente ausreichend klein sein, ist dieser Ansatz zur Näherung ebenfalls zulässig.

Eine weitere Möglichkeit der Substrukturierung beschreiben SCHAUER und LEHMANN [139]. Sie untersuchen den Einfluss elementweise überlappender Substrukturen, um die Größe der einzelnen Substrukturen so klein wie möglich zu halten. Dieser Ansatz wird hier nicht weiter betrachtet, da die Netzerzeugung aufwändig ist und keinen wesentlichen Geschwindigkeitsvorteil im Verhältnis zum relativen Fehler der Lösung bringt.

Die Untersuchungen im vorangegangenen Abschnitt haben gezeigt, dass der benötigte Speicher durch eine geometrische Entkopplung drastisch reduziert werden kann, die Rechenzeit aufgrund der konstanten Systemgröße allerdings nahezu konstant bleibt. Wie sich die Substrukturierung des Fernfeldes auf Speicherbedarf und Rechenzeit sowie das Lösungsverhalten auswirkt, wird nachfolgend genauer betrachtet.

Zur Untersuchung der geometrischen Entkopplung wird hier das Fernfeld HSM 9 in Substrukturen zerlegt, wodurch jede Substruktur lediglich einen Teil des gesamten Fernfeldes repräsentiert. Durch dieses Vorgehen wird die Systemgröße des Fernfeldes reduziert, da jedes Subsystem separat und unabhängig von den anderen Substrukturen diskretisiert wird. Erst bei der gekoppelten Simulation von Nah- und Fernfeld werden die Substrukturen wieder zusammengesetzt. Mit der Zerlegung des Fernfeldes in n Substrukturen werden innerhalb des Fernfeldes künstliche Ränder geschaffen, die in der Realität nicht auftreten. Der Einfluss der künstlichen Ränder wird nachfolgend untersucht.

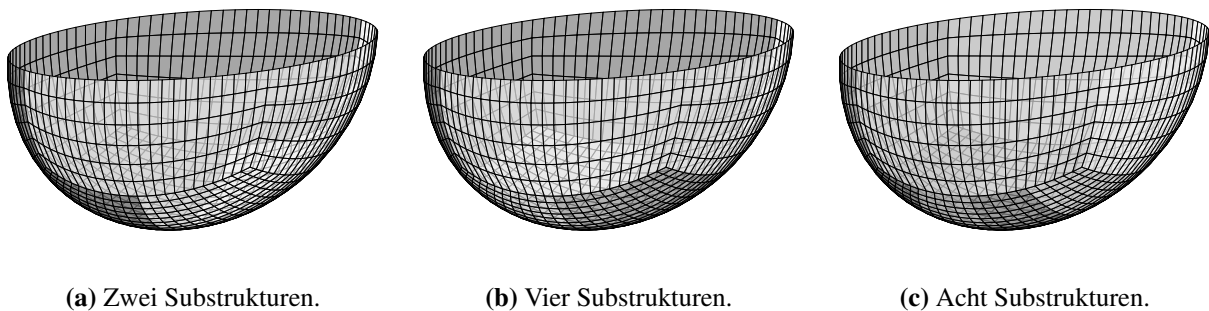


Bild 6.7.: Zerlegung des Fernfeldes in zwei (a), vier (b) und acht (c) Substrukturen am Beispiel des HSM 9.

Die Bilder 6.7a bis 6.7c zeigen exemplarisch die Zerlegung des Fernfeldes in zwei, vier und acht Substrukturen anhand des HSM 9 Fernfeldes. Die Zerlegung erfolgt hier auf Basis der Elemente, sodass jeder Substruktur die gleiche Anzahl an Elementen zugewiesen wird. Die Anzahl der Freiheitsgrade jeder Substruktur ist nicht zwangsläufig identisch und variiert somit trotz gleicher Anzahl an Elementen je nach Geometrie leicht.

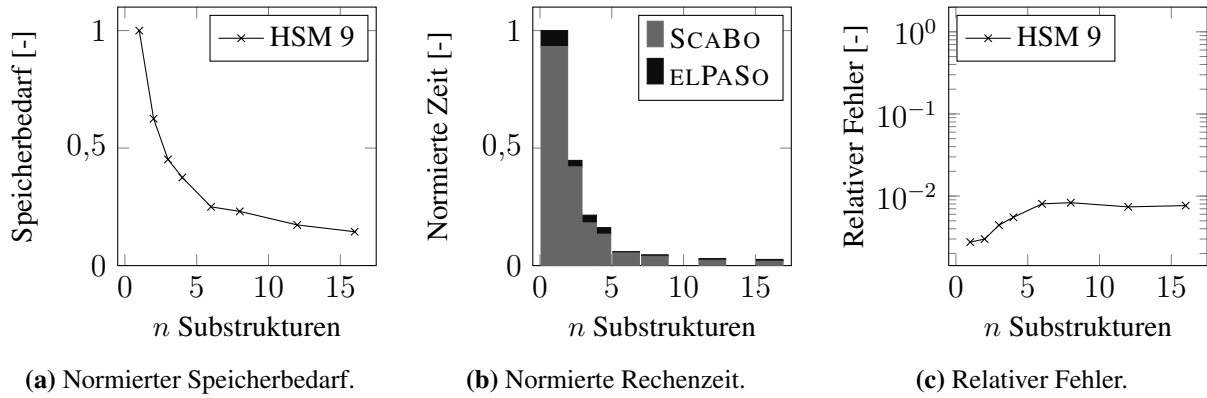


Bild 6.8.: Speicherbedarf der Einflussmatrizen M^∞ (a), normierte Rechenzeit (b) sowie der relativer Fehler der Setzungsberechnung (c) in Abhängigkeit der Anzahl an Substrukturen für das Netz HSM 9.

Mit zunehmender Anzahl an Substrukturen nehmen Speicherbedarf und Rechenzeit deutlich ab. Die Bilder 6.8a und 6.8b zeigen die Summe des Speicherbedarfs beziehungsweise der Rechenzeit aller n Substrukturen. In Bild 6.8c ist der relative Fehler der Setzungssimulation in Abhängigkeit der Anzahl von Substrukturen dargestellt. Es wird deutlich, dass der Fehler im Verhältnis zu den vorangegangenen Verfahren zur Effizienzsteigerung größer ausfällt. Bild 6.8c zeigt einen deutlichen Anstieg des relativen Fehlers im Bereich von zwei bis sechs Substrukturen. Danach bleibt der Fehler nahezu konstant. Der große Vorteil der Zerlegung des Fernfeldes in Substrukturen ist der massiv reduzierte Speicherbedarf sowie die ebenfalls massiv reduzierte Rechenzeit (vgl. Bild 6.8a und 6.8b). Eine derartige Reduktion von Speicherbedarf und Rechenzeit kann mit der Extrapolation der Einflussmatrizen oder der geometrischen Entkopplung nicht realisiert werden.

Werden alle drei Modellreduktionen miteinander kombiniert, lassen sich Speicher- und Rechenaufwand weiter reduzieren. Eindrucksvoll zeigen lässt sich dies an dem folgenden Beispiel: Wird ein Fernfeld bestehend aus vier Substrukturen gewählt und zusätzlich für die Extrapolation $t_m = 50$ sowie $\epsilon_z = 1,0 \cdot 10^{-5}$ für die geometrische Entkopplung eingeführt, dann folgt bei gleicher relativer Fehlergröße, verglichen mit dem System aus vier Substrukturen ohne weitere Modellreduktionen, eine Rechenzeit von gerade einmal 6,1% im Vergleich zur Referenzlösung (vollständiges Fernfeld, $t_m = 50$, $\epsilon_z = 1,0 \cdot 10^{-5}$). Auch hier bildet das Produkt der einzelnen prozentualen Rechenzeiten (Extrapolation $t_m = 50$ 36,4%, geometrische Entkopplung $\epsilon_z = 1,0 \cdot 10^{-5}$ 90,6% und Substrukturierung in vier Teilsysteme 16,2%) in etwa den Anteil der gesamten Rechenzeit der kombinierten Modellreduktionen, sodass der Speicherbedarf bei etwa 11% gegenüber der Referenzlösung liegt.

6.4. Fazit

Wie schon in Abschnitt 5.2.2 gezeigt, ist die Genauigkeit der HSM besser als die der CSM, was durch die Betrachtung des relativen Fehlers eindeutig zu belegen ist. Dies gilt ebenfalls, wenn eine der vorgestellten Modellreduktionen zum Lösen des Setzungsproblems gewählt wird. Die CSM-Netze sind hinsichtlich der Modellreduktion robuster als die HSM. Diese Beobachtung deckt sich mit den Ergebnissen von BORSUTZKY [32], der ebenfalls Untersuchungen zur Modellreduktion anhand rechteckiger Fernfelder beschrieben hat.

Die Extrapolation der Einflussmatrizen ist ein sehr leistungsfähiges Verfahren, mit Hilfe dessen sowohl der benötigte Speicher als auch die zum Lösen des Problems erforderliche Rechenzeit stark reduziert werden können. Dies ist jedoch nur dann möglich, wenn der Zeitpunkt t_m der Extrapolation korrekt gewählt wird. Ziel ist es, mit der Extrapolation frühestmöglich zu beginnen. Es muss allerdings beachtet werden, dass die Verwendung von wenigen Einflussmatrizen zu großen Fehlern in der Simulation führen kann. Um den Extrapolationsansatz nutzen zu können, ist der Algorithmus zum Lösen des gekoppelten Problems aus Nah- und Fernfeld, wie in Gleichung (6.3) bis (6.8) gezeigt, mit einem rekursiven Ansatz entsprechend anzupassen.

Mit dem Einsatz der geometrischen Entkopplung kann der Speicherbedarf drastisch reduziert werden. Auf die erforderliche Rechenzeit ist der Einfluss eher gering, da nach wie vor dasselbe Fernfeld berücksichtigt wird. Lediglich beim Speichern und Einlesen der Einflussmatrizen folgt aufgrund der reduzierten Datenmenge ein zeitlicher Vorteil. Das Verfahren ist im Vergleich zu den beiden anderen vorgestellten Verfahren das am einfachsten programmtechnisch umzusetzende, da ausschließlich die Routine zum Schreiben der Einflussmatrizen anzupassen ist. Diese muss sicherstellen, dass nur Werte größer ϵ_z berücksichtigt werden.

Die effiziente Nutzung der Substrukturierung erfordert eine Anpassung der Kopplung von Nah- und Fernfeld, da das Nahfeld mit n Substrukturen des gesamten Fernfeldes gekoppelt wird. Des Weiteren muss das Fernfeld vom Nutzer in Substrukturen zerlegt werden, wodurch der Aufwand der Netzgenerierung zunimmt. Der höhere Aufwand in der Netzerzeugung zahlt sich jedoch aus, da Speicherbedarf und Rechenzeit drastisch reduziert werden.

Die Kombination aller drei Verfahren der Modellreduktion liefert die größten Speicher- und Rechenzeiterparnisse. Extrapolation der Einflussmatrizen, geometrische Entkopplung und Substrukturierung können, wie gezeigt, zu einer erheblichen Reduktion der Rechenzeit um etwa 94% führen. Dabei ist zu beachten, dass der größte relative Fehler, der in einer der Modellreduktionen auftritt, den limitierenden Faktor hinsichtlich der Rechengenauigkeit darstellt. Der relative Fehler des gesamten Systems ist also mindestens so groß wie der größte relative Fehler einer einzelnen Modellreduktion. Dies ist bei der Nutzung der Strategien zur Effizienzsteigerung unbedingt zu beachten.

7. Parallelisierung der diskretisierten Modellgleichungen

Ein wichtiges Qualitätsmerkmal paralleler Software ist deren Skalierbarkeit, da diese Aussagen bezüglich der Leistungsfähigkeit ermöglicht. Die Leistungsfähigkeit wird über zwei wesentliche Kenngrößen, den *Speedup* sowie die Effizienz, dargestellt. Auf diese beiden Kenngrößen wird im Weiteren detailliert eingegangen. Der grundlegende Ansatz stammt von AMDAHL [1,2] aus dem Jahr 1967, der von GUSTAVSON [74] 1988 erweitert wird. Weitere Möglichkeiten zur Bestimmung der Kenngrößen unter Berücksichtigung des zur Verfügung stehenden Arbeitsspeichers je Prozess werden von SUN und NI [155] sowie SUN und CHEN [154] diskutiert. Auf diesen *Shared Memory Parallel* SMP Ansatz, der sich auf *Multicore* Systeme bezieht, wird im Weiteren nicht eingegangen. Im Nachfolgenden wird der *Distributed Memory Parallel* DMP Ansatz verfolgt, da der zur Untersuchung verwendete Rechencluster über jeweils eine Recheneinheit je CPU (*Singlecore*) verfügt.

7.1. Amdahls und Gustafsons Gesetze

Es wird angenommen, dass sich die Gesamtrechenzeit T eines Rechenschemas aus einem sequentiellen Anteil αT und einem parallelisierbaren Anteil $(1 - \alpha) T$ zusammensetzt. Zu dem sequentiellen Anteil zählt in der Regel das Lesen der Eingabedatei, ggf. das Schreiben der Ergebnisse sowie Ausgaben für den Nutzer. Hinzu kommen zusätzliche Anweisungen, die für die Kommunikation zwischen den einzelnen Prozessen erforderlich sind. Es gilt somit:

$$T = \alpha T + (1 - \alpha) T. \quad (7.1)$$

Wird dieser Ansatz auf p Prozesse erweitert folgt:

$$T(p) = \alpha T(1) + \frac{(1 - \alpha) T(1)}{p}. \quad (7.2)$$

Der Quotient aus sequentieller Gesamtrechenzeit $T(1)$ und paralleler Rechenzeit $T(p)$ ist der *Speedup*, der mit

$$S_p = \frac{T(1)}{T(p)} = \frac{\alpha T(1) + (1 - \alpha) T(1)}{\alpha T(1) + \frac{(1 - \alpha) T(1)}{p}} = \frac{\alpha + (1 - \alpha)}{\alpha + \frac{(1 - \alpha)}{p}} = \frac{1}{\alpha + \frac{(1 - \alpha)}{p}} \quad (7.3)$$

definiert ist. Der *Speedup* gibt an, um welchen Faktor die numerische Simulation beschleunigt wird. Ein *Speedup* von Zwei bedeutet, dass sich die erforderliche Rechenzeit halbiert. Mit Gleichung (7.3) kann dann mit Hilfe einer Grenzwertbetrachtung der maximal erreichbare *Speedup* $S_{p,\max}$ bestimmt werden. Dabei wird davon ausgegangen, dass unendlich viele Prozesse zur Verfügung gestellt werden können. Daraus folgt, dass der maximale *Speedup*

$$S_{p,\max} = \lim_{p \rightarrow \infty} \frac{1}{\alpha + \frac{(1 - \alpha)}{p}} = \frac{1}{\alpha} \quad (7.4)$$

durch den sequentiellen Programmteil begrenzt ist. Der theoretische *Speedup* $S_p = p$ für $\alpha = 0$ wird nie erreicht, da dies die vollständige Parallelisierung einer Berechnung bedeuten würde und alle an der Berechnung beteiligten Prozesse unabhängig voneinander wären. Eine Kommunikation zwischen den Prozessen wäre in diesem Fall nicht möglich und ist auch nicht vorgesehen.

Wird der parallele *Speedup* durch die Anzahl der Prozesse p geteilt, kann die parallele Effizienz (*parallel efficiency*)

$$E_p = \frac{S_p}{p} = \frac{T(1)}{p \cdot T(p)} \quad (7.5)$$

des Programms bestimmt werden. Diese beschreibt die Effizienz jedes einzelnen Prozesses. In der Regel fällt die Effizienz der beteiligten Prozesse mit deren Anzahl, da mit jedem zusätzlichen Prozess weitere Kommunikation hinzukommt. Wird die Effizienz für p Prozesse mit beispielsweise 80% angegeben, dann bedeutet dies, dass jeder Prozess 80% der Rechenzeit zum Lösen des gegebenen Problems einsetzt. Die restlichen 20% der Rechenzeit verbringt der Prozess entweder mit der Synchronisation oder mit der Kommunikation (vgl. Abschnitt 4.1).

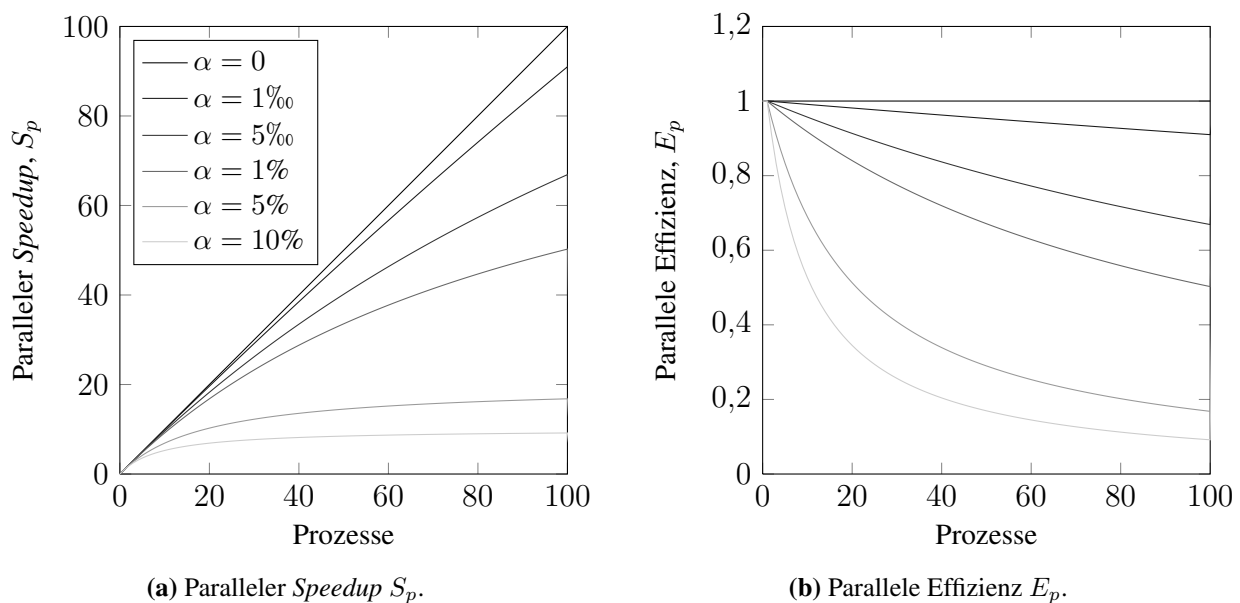


Bild 7.1.: *Speedup* und Effizienz in Abhängigkeit des sequentiellen Programmanteils α .

Die Bilder 7.1a und 7.1b zeigen den parallelen *Speedup* und die parallele Effizienz für unterschiedliche sequentielle Anteile α sowie das theoretisch erreichbare Maximum, das in diesem Fall für $\lim_{\alpha \rightarrow 0} S_p = p$ und $E_p = 1$ folgt. Die Grafiken zeigen die Limitierung des *Speedups*, die sich aufgrund des sequentiellen Anteils ergibt. Bei einem sequentiellen Anteil von lediglich 10% ist ein maximaler *Speedup* von 10 theoretisch erreichbar, wenn unendlich viele Prozesse parallel gestartet werden könnten. Wird der sequentielle Anteil auf 5% reduziert, ist ein maximaler *Speedup* von 20 unter gleichen Voraussetzungen theoretisch möglich. Soll beispielsweise

bei der Nutzung von 50 Prozessen die parallele Effizienz bei mindestens 50% liegen, darf der sequentielle Anteil der Berechnung 4% an sequenziellen Programmaufrufen nicht überschreiten. In diesem Fall folgt ein *Speedup* von 25. Folglich ist das Ziel, den sequentiellen Anteil α so klein wie möglich zu halten, um eine möglichst leistungsstarke Software bereitzustellen.

Anders als AMDAHL setzt GUSTAFSON nicht eine von vornherein festgelegte Problemgröße voraus, die auf p Prozesse aufgeteilt wird, sondern ein festes Zeitfenster, in dem eine Fragestellung abgearbeitet werden muss [74]. Dabei verändert er die Problemgröße proportional in Abhängigkeit der Anzahl an beteiligten Prozessen, um den skalierten *Speedup* S'_p zu bestimmen. Die grundlegende Idee dahinter ist, in einer definierten Zeitspanne T' ein Problem zu untersuchen und mit Hilfe zusätzlicher Prozesse nicht die Rechenzeit zu reduzieren, sondern einen höheren Detailgrad abzubilden.

Wie schon bei AMDAHL gilt auch hier, dass sich die gesamte Rechenzeit T' aus zwei Anteilen zusammen setzt: aus einem sequentiellen Anteil α und einem parallelen Anteil $(1 - \alpha)$, die in der Zeit T ablaufen. Es gilt somit

$$T' = \alpha T + (1 - \alpha) p T \quad (7.6)$$

(vgl. Gleichung (7.1)). In Gustafsons Ansatz ist der parallele Anteil mit der Anzahl an gestarteten Prozessen gewichtet. Somit ergibt sich der skalierte *Speedup* zu

$$S'_p = \frac{T'(1)}{T'(p)} = \frac{\alpha T + (1 - \alpha) p T}{\alpha T + \frac{(1 - \alpha) p T}{p}} = p(1 - \alpha) + \alpha. \quad (7.7)$$

Damit wird deutlich, dass die Limitierung des *Speedups* durch den sequentiellen Anteil aus Gleichung (7.4) entfällt. Der maximale skalierte *Speedup* beruht nun auf der maximal verfügbaren Anzahl an Prozessen und skaliert bei ausreichender Systemgröße linear mit der Anzahl an Prozessen. Die Effizienz der einzelnen Prozesse wird analog zu Amdahls Gesetz mit

$$E'_p = \frac{S'_p(p)}{p} \quad (7.8)$$

bestimmt.

Die Skalierbarkeit einer Software lässt sich daher mit zwei unterschiedlichen Tests bestimmen. Zum einen mit der starken Skalierung (*strong scaling*) nach AMDAHL. Dabei wird ein definierter Testfall mit Hilfe des zu testenden Programms sequentiell gelöst und die erforderliche Zeit $T(1)$ ermittelt. Im Anschluss daran erfolgt das Lösen desselben Testfalls mit Hilfe von p parallelen Prozessen, wobei für jeden Lauf die benötigte Zeit $T(p)$ ermittelt wird. Aus den gemessenen Zeiten lassen sich für jede beliebige Anzahl von Prozessen zwei allgemeingültige Größen ermitteln. Zum einen gibt der *Speedup* S_p , wie in Gleichung (7.3) beschrieben, das zeitliche Verhältnis zwischen einem und p Prozessen wieder, wobei die parallele Effizienz (vgl. Gleichung (7.5)) die Effizienz der beteiligten Prozesse darstellt.

Zum anderen kommt das *weak scaling* nach GUSTAFSON zum Einsatz. Hierbei wird ein Testfall in Abhängigkeit der Anzahl der parallelen Prozesse skaliert, sodass jeder Prozess unabhängig

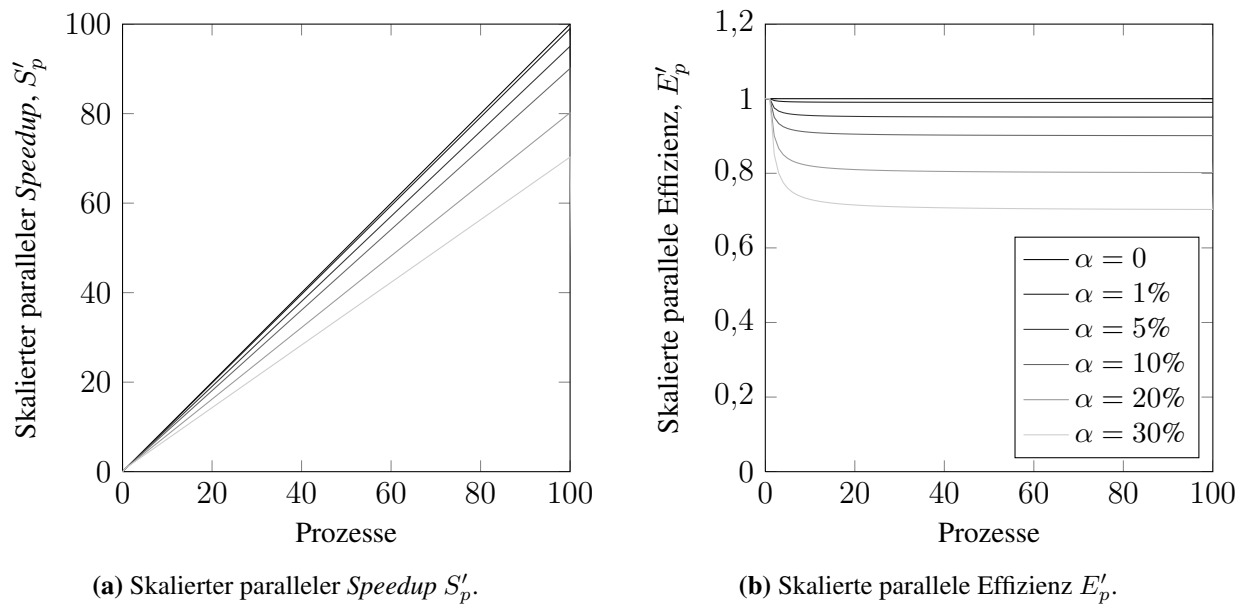


Bild 7.2.: Skalierter *Speedup* und skalierte Effizienz in Abhängigkeit des sequentiellen Programmanteils α .

von der Anzahl der Gesamtprozesse die gleiche Problemgröße lösen muss. Die Bestimmung des skalierten *Speedup* S'_p sowie der skalierten parallelen Effizienz E'_p ist in diesem Fall wesentlich aufwändiger und zumeist problemabhängig.

Welcher der beiden Tests wichtiger oder aussagekräftiger ist, hängt immer von der Fragestellung ab. Wenn es darum geht, ein bestimmtes Problem n -mal zu lösen, dann ist es oftmals sinnvoll, im Vorfeld eine Untersuchung bezüglich des starken Skalierungsverhaltens durchzuführen, um eine optimale Anzahl von parallelen Prozessen zu ermitteln. Sollte es sich jedoch beispielsweise um Konvergenzuntersuchungen oder Probleme mit wachsender Systemgröße handeln, muss das Programm im Hinblick auf das schwache Skalierungsverhalten untersucht werden. Da die Problemgröße der heutigen Rechenmodelle in der Regel stetig ansteigt, ist es notwendig Programme zu entwickeln, die bei der schwachen Skalierung überzeugen können.

Alle nachfolgenden Untersuchungen zur parallelen Leistung der implementierten Software werden auf einem Computercluster durchgeführt (AMAN PC-Cluster [80]). Jeder der zur Zeitmessung herangezogenen Rechenknoten ist ausgestattet mit zwei *AMD Opteron-240* Prozessoren, die mit 1,4 GHz getaktet sind. Jeder Prozessor verfügt über drei Gigabyte Arbeitsspeicher. Die Kommunikation zwischen den Rechenknoten übernimmt ein *Myrinet-2000 Interconnector*. Für die Messungen wird je Prozessor maximal ein MPI Prozess gestartet. Die Rechenknoten werden erst aufgefüllt, bevor ein weiterer Knoten hinzugenommen wird. Ein Testlauf mit zwei Prozessen läuft dementsprechend auf nur einem Knoten.

7.2. SCABO: Fernfeldberechnung

Ein Teil der Ergebnisse ist bereits von SCHAUER et al. [143] veröffentlicht und ausführlich diskutiert. Eine detaillierte Betrachtung der einzelnen Programmabschnitte erfolgt hier nicht, es wird lediglich auf die Rechenzeit eingegangen. Als Grundlage der nachfolgenden Betrachtung des gekoppelten Ansatzes werden die Ergebnisse kurz zusammengefasst. Die Bewertung der Parallelisierung erfolgt hier sowohl anhand der starken als auch der schwachen Skalierung. Bei der starken Skalierung werden die Rechenzeiten für ein Problem konstanter Größe auf unterschiedlich vielen Prozessen gemessen, bei der schwachen Skalierung wächst die Problemgröße proportional mit der Anzahl der Prozesse wie bereits in diesem Kapitel eingangs beschrieben. Die aktuelle Version von SCABO ermöglicht bei der Berechnung des n -ten Zeitschrittes, die rechte Seite der Gleichung (3.68) unter Berücksichtigung symmetrischer Matrizen zu lösen. Berechnungen, die im Lösungsprozess die Symmetrie der Einflussmatrizen berücksichtigen, sind entsprechend gekennzeichnet.

7.2.1. Starke Skalierung

Für die Untersuchung der Parallelisierung wird ein relativ kleines SBFEM Netz mit 505 Knoten gewählt. Daraus folgt ein Gleichungssystem mit 1515 Unbekannten (vgl. Netz HSM 5, Kapitel 5.2.2, Tabelle 5.4). Die Anzahl der zu bestimmenden M_n^∞ Matrizen ist auf 100 begrenzt, wodurch sichergestellt ist, dass das gewählte Problem in angemessener Zeit mit einem Prozess gelöst werden kann und der zur Verfügung stehende Arbeitsspeicher nicht überschritten wird. Durch Auswerten der Zeitmessung entsprechend den Gleichungen (7.3) und (7.5) lassen sich *Speedup* und Effizienz des implementierten Algorithmus bestimmen. Die Ergebnisse der Zeitmessungen sowie die Ergebnisse aus der bereits erwähnten Veröffentlichung sind in den Abbildungen 7.3a und 7.3b zusammengefasst. Die Zeitmessungen von SCHAUER et al. erfolgen anhand eines in der Größenordnung vergleichbaren SBFEM Netzes mit 409 Knoten und 1227 Unbekannten (vgl. Netz CSM 3, Kapitel 5.2.2, Tabelle 5.3).

Es wird deutlich, dass der *Speedup* bei allen drei Messreihen mit zunehmender Anzahl an Prozessen stetig zunimmt. Der theoretisch mögliche *Speedup* wird jedoch nur mit einer kleinen Zahl an Prozessen näherungsweise erreicht. Je größer die Anzahl der eingesetzten Prozesse, desto größer ist der Abstand zu dem theoretischen Maximum. Bei 48 Prozessen liegt der gemessene *Speedup* lediglich bei 16,2 bis 27,5. Die parallele Effizienz nimmt, wie zu erwarten, mit zunehmender Anzahl an Prozessen stetig ab, bei 48 gestarteten Prozessen liegt die Effizienz bei lediglich 33,8% bis 57,3%.

Wie zu erwarten, ist die parallele Effizienz im Falle dieser kleinen Beispiele nicht besonders gut, da die lokale Problemgröße mit zunehmender Anzahl an Prozessen stark abnimmt. Gleichzeitig nimmt die benötigte Zeit für die Kommunikation zwischen den Prozessen zu und wird nicht zum Lösen des eigentlichen Problems genutzt. SCHAUER et al. [143] zeigen, dass mit zunehmender

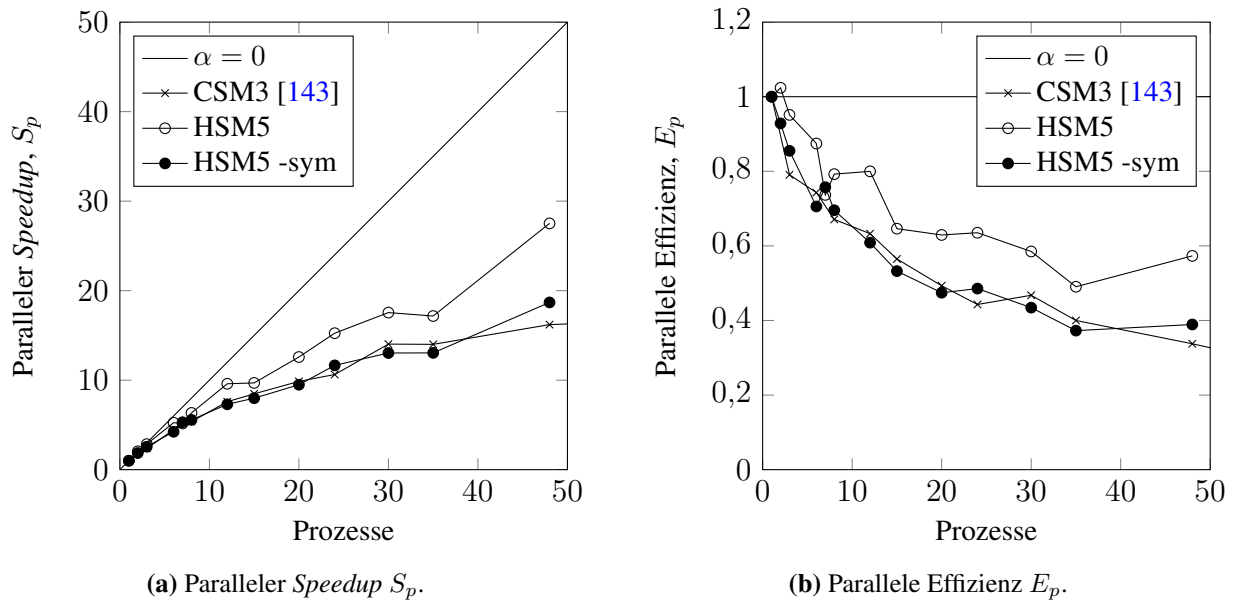


Bild 7.3.: SCABO: Paralleler Speedup sowie parallele Effizienz der starken Skalierung.

Systemgröße die parallele Effizienz verbessert wird. Derselbe Trend ist hier zu beobachten, da das Netz HSM 5 mit 1515 Unbekannten größer ist als das Netz CSM 3 mit 1227 Unbekannten.

7.2.2. Schwache Skalierung

Die Anwendbarkeit der SBFEM für großskalige Probleme mit vielen Freiheitsgraden wird anhand der schwachen Skalierung gezeigt. In diesem Fall steigt die Anzahl der Freiheitsgrade proportional mit der Anzahl der Prozesse, sodass die lokale Problemgröße in jedem Prozess nahezu konstant bleibt. Die Problemgröße wird so ausgelegt, dass jeder Prozess eine Untermatrix mit etwa 1,2 Millionen Einträgen zu bearbeiten hat. Bei den Netzen handelt es sich um nahezu quadratische Netze, die hier für den *Benchmark* eingesetzt werden. In Tabelle 7.1 sind die untersuchten Netze, der jeweiligen Anzahl an Prozessen zugeordnet, zusammengefasst. Zusätzlich zu der Anzahl an Knoten und der daraus resultierenden Anzahl an Freiheitsgraden sind ebenfalls die Gesamtanzahl der Matrixeinträge und die damit verbundene Anzahl der Matrixeinträge je Prozess aufgeführt. Je nach Netz und Anzahl der zugeordneten Prozesse schwankt die Zahl der Untermatrixeinträge (N/p) um -1,93% bis 3,99% im Vergleich zum Referenzwert $N/p = 1200000$. Die verwendeten Netze entsprechen denen aus der Veröffentlichung von SCHAUER et al. [143].

Netz	Prozesse p	Knoten	Freiheitsgrade	Matrizeinträge N	N/p
19×19	1	361	1083	1172889	1172889
22×23	2	506	1518	2304324	1152162
25×25	3	625	1875	3515625	1171875
30×30	6	900	2700	7290000	1215000
32×32	8	1024	3072	9437184	1179648
35×36	12	1260	3780	14288400	1190700
37×38	15	1406	4218	17791524	≈ 1186102
40×41	20	1640	4920	24206400	1210320
42×43	24	1806	5418	29354724	≈ 1223114
44×45	30	1980	5940	35283600	1176120
46×47	35	2162	6486	42068196	≈ 1201949
48×49	42	2352	7056	49787136	1185408
52×53	56	2756	8268	68359824	≈ 1220711

Tabelle 7.1.: Netze für die schwache Skalierung. Die Netzgröße ist proportional zur Anzahl der Prozesse.

Ebenso wie bei der starken Skalierung wird auch bei der schwachen Skalierung die Leistungsfähigkeit mit dem parallelen *Speedups* sowie der parallelen Effizienz beurteilt. Der skalierte *Speedup*

$$S'_p = \frac{p\sqrt{p} \cdot T(1)}{T'(p)} \quad (7.9)$$

und die skalierte Effizienz

$$E'_p = \frac{\sqrt{p} \cdot T(1)}{T'(p)} \quad (7.10)$$

werden analog zu [143] bestimmt. Es wird angenommen, dass die einzelnen Prozesse p in einem perfekten Quadrat $\sqrt{p} \times \sqrt{p}$ angeordnet werden können und jede lokale Untermatrix $n \times n$ Einträge enthält, wobei $n = \sqrt{N}$ entspricht. $T(1)$ beschreibt somit die Zeit, die zum Lösen eines Problems der Größe n auf einem Prozess benötigt wird. Daraus folgt, dass bei der Betrachtung der schwachen Skalierung Systeme mit der Größe $n\sqrt{p}$ auf p Prozessen untersucht werden können. Die dafür erforderliche Rechenzeit ist $T'(p)$. Der *Speedup* wird wie zuvor aus dem Verhältnis von $T'(1)$ und $T'(p)$ bestimmt. Die Rechenzeit $T'(1)$ für ein System der Größe $n\sqrt{p}$ auf einem Prozess ist unbekannt und wird mit $T(1)$ abgeschätzt. Da die Komplexität des Algorithmus im Wesentlichen kubisch ist und bei einer Systemgröße von $n\sqrt{p}$ der Ordnung $O(n^3 \cdot p\sqrt{p})$ entspricht, ist der parallele *Speedup* entsprechend Gleichung (7.9) zu skalieren. Konsequenterweise erfolgt die Ermittlung der skalierten Effizienz in Gleichung (7.10).

Die skalierten Größen sind in den Bildern 7.4a und 7.4b dargestellt. Es wird deutlich, dass die Ergebnisse der schwachen Skalierung wesentlich besser ausfallen als die der starken Skalierung. Die parallele Effizienz des gesamten Algorithmus liegt bei etwa 82,5% bis 97,8%. Die

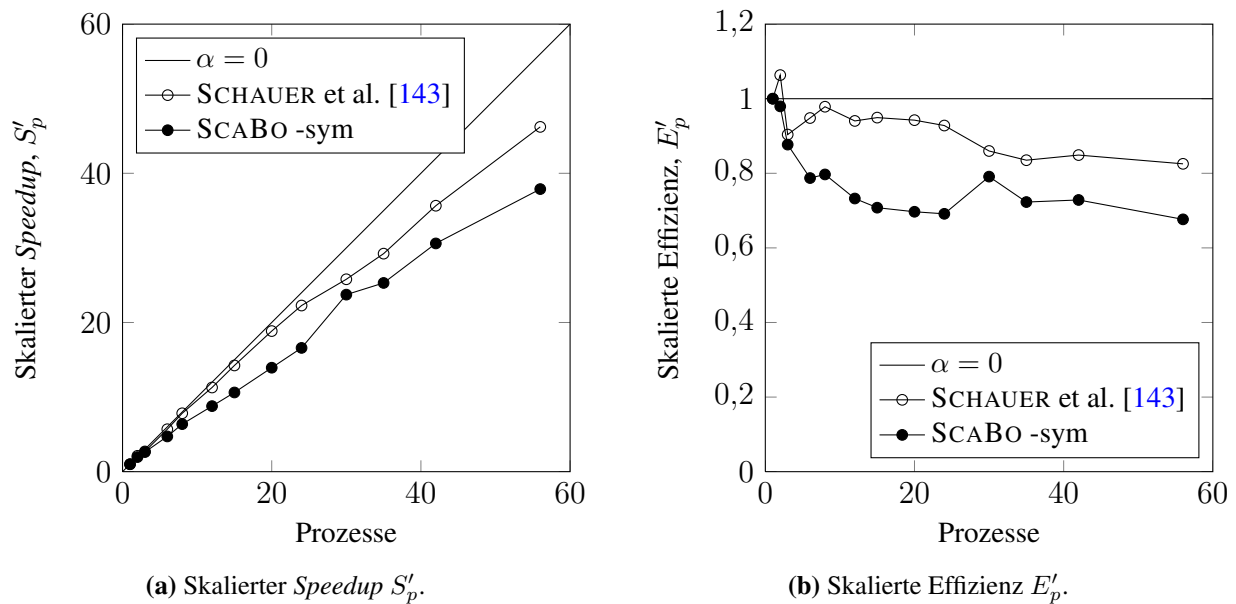


Bild 7.4.: SCABO: Skalierter *Speedup* sowie skalierte Effizienz der schwachen Skalierung.

Effizienz nimmt mit steigender Anzahl an Prozessen leicht ab, was auf die zusätzlich Kommunikation zwischen den Prozessen zurückzuführen ist. Wird die Symmetrie der Matrizen bei der Berechnung berücksichtigt, ist eine Effizienz von immerhin 69,1% bis 97,9% zu verzeichnen und fällt somit etwas schlechter aus.

7.2.3. Vergleich der Lösungsansätze

In den beiden vorherigen Abschnitten werden der parallele *Speedup* sowie die parallele Effizienz der Software SCABO für zwei unterschiedliche Lösungsansätze diskutiert, auf die nun näher eingegangen wird. Beide Lösungsansätze entsprechen dem in Kapitel 3.3.2 vorgestellten und in Kapitel 4.2.1 näher erläuterten Lösungsweg. Die mit „-sym“ gekennzeichneten Graphen beziehen sich auf einen Lösungsansatz, bei dem die Symmetrie der Matrizen während der Berechnung berücksichtigt wird. Der symmetrische Lösungsansatz nutzt die Symmetrie der Einflussmatrix beim Lösen der Lyapunov Gleichung (vgl. (3.71)) jedes n -ten Zeitschrittes aus. Dies bedeutet, dass lediglich die Berechnung der Gleichungen (3.69) und (3.70) sowie die Bestimmung der rechten Seite von Gleichung (3.68) verändert sind. Alle anderen Programmteile bleiben unverändert bestehen.

Werden bei der starken sowie schwachen Skalierung lediglich *Speedup* und Effizienz betrachtet, schneidet der Algorithmus, der die Symmetrie der Matrizen berücksichtigt, schlechter ab als der des herkömmlichen Lösungsverfahrens. Wird jedoch zusätzlich die tatsächlich benötigte Rechenzeit beider Lösungsansätze berücksichtigt, muss die Bewertung anders ausfallen. Der Quotient ψ wird genutzt, um die für die vollständige Berechnung benötigte Rechenzeit in beiden

Fällen ins Verhältnis zueinander zu setzen. Im Falle der starken Skalierung wird dieser mit

$$\psi = \frac{T(p)_{\text{sym}}}{T(p)} \quad (7.11)$$

bestimmt. Für die schwache Skalierung gilt:

$$\psi' = \frac{T'(p)_{\text{sym}}}{T(p)}. \quad (7.12)$$

Beide Quotienten lassen sich dann über die Anzahl der Prozesse auftragen, wie im Bild 7.5 dargestellt.

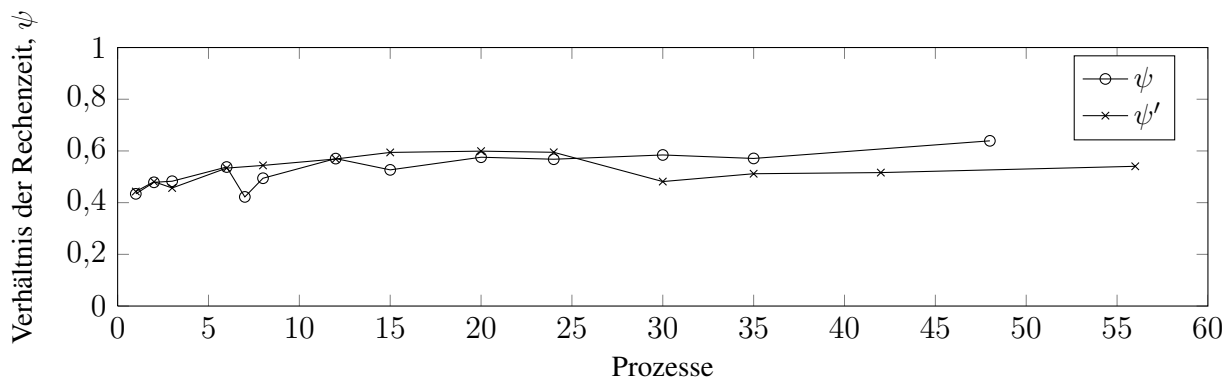


Bild 7.5.: Rechenzeit des herkömmlichen und des symmetrischen Lösungsansatzes für die starke (ψ) sowie schwache (ψ') Skalierung.

Je nach Konfiguration benötigt der neue symmetrische Lösungsansatz lediglich 42% bis 64% der ursprünglichen Rechenzeit und ist dem anderen somit vorzuziehen. Wie bereits erwähnt, hat der symmetrische Lösungsansatz lediglich einen Einfluss auf die Berechnung der rechten Seite der Gleichung (3.68). Wird der Quotient ausschließlich für diesen Teil der Simulation ermittelt, folgt eine Reduktion der Rechenzeit im Falle der starken Skalierung auf 39,9% bis 58,8% der ursprünglich benötigten Zeit. Daraus lässt sich ableiten, dass die Effizienz des symmetrischen Lösungsansatzes mit zunehmender Anzahl an Zeitschritten ansteigt. Auch bei der schwachen Skalierung verkürzt sich die Rechenzeit auf 44,3% bis 59,5% der ursprünglich benötigten Zeit je nach Anzahl an Prozessen. Der Vorteil des symmetrischen Lösungsansatzes liegt somit auf der Hand und ist dem ursprünglich in SCABO implementierten Ansatz vorzuziehen.

7.3. ELPASO: Gekoppelter-Nahfeld-Fernfeld-Ansatz

Die Effizienzanalyse des gekoppelten Ansatzes aus FEM und SBFEM erfolgt anhand der starken Skalierung. Eine Untersuchung auf Grundlage der schwachen Skalierung ist kaum möglich, da bei einer Netzverfeinerung die Zahl der Freiheitsgrade im Nahfeld und Fernfeld unterschiedlich stark anwächst. Die Anzahl der Freiheitsgrade im Nahfeld wächst kubisch, die im Fernfeld hingegen lediglich quadratisch (vgl. Bild 5.13).

Für die starke Skalierung wird das Beispiel HSM 9 (vgl. Abschnitt 5.2.2) auf unterschiedlicher Anzahl von Prozessen gelöst. Das hier gewählte Beispiel besteht aus 49155 Freiheitsgraden im Nahfeld (FEM), hinzu kommen weitere 3603 Freiheitsgrade des Fernfeldes (SBFEM). In dem hier diskutierten Beispiel werden 100 Zeitschritte ausgewertet, die Anzahl der Einflussmatrizen ist auf 50 Matrizen limitiert, sodass eine Extrapolation der Einflussmatrizen nach dem fünfzigsten Zeitschritt erfolgt (vgl. Abschnitt 6.1). Die Toleranz der geometrischen Entkopplung ϵ_z ist zu $1,0 \times 10^{-7}$ gewählt. Eine Zerlegung des Fernfeldes, wie in Abschnitt 6.3 beschrieben, findet nicht statt. Untersucht wird der gesamte Algorithmus des gekoppelten Nahfeld-Fernfeld-Ansatzes bestehend aus Gleichungslöser, Kopplungsalgorithmus und Zeitschrittintegrationsverfahren. Eine detaillierte Unterteilung erfolgt an dieser Stelle nicht, es wird lediglich die Zeit, die zum Lesen der Einflussmatrizen erforderlich ist, herausgerechnet. Dies ist notwendig, da nicht sichergestellt werden kann, dass jede Simulation auf dem Cluster exklusiv auf das Dateisystem zugreifen kann und somit unterschiedliche Simulationen, die zeitgleich auf dem System laufen, sich für die Zeitmessung gegenseitig negativ beeinflussen können. Lese- und Schreibzugriffe auf das Dateisystem können daher in ihrer Geschwindigkeit je nach Situation sehr unterschiedlich ausfallen. Als Referenzwert zur Ermittlung von *Speedup* und Effizienz dient auch hier wie schon zuvor bei der Effizienzanalyse der SBFEM die sequentielle Rechenzeit T_1 . Das Problem wird mit einem bis zu 30 Prozessen gelöst. Auch hier werden die Rechenknoten erst aufgefüllt, bevor ein weiterer Knoten hinzugefügt wird, sodass maximal zwei MPI-Prozesse je Rechenknoten laufen.

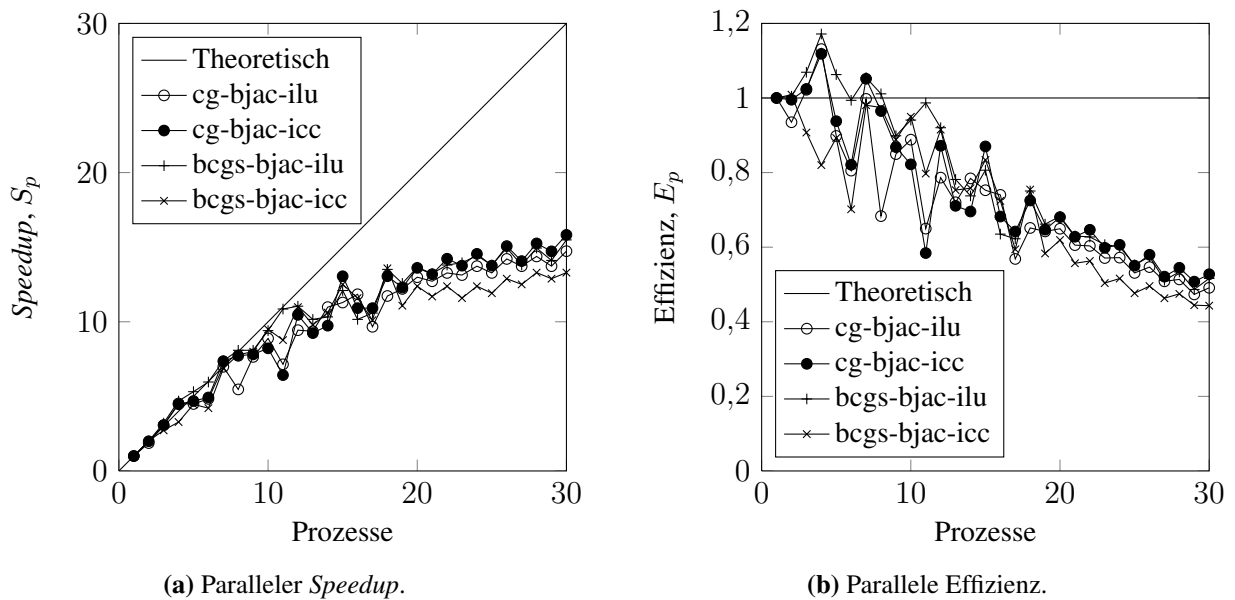


Bild 7.6.: ELPASO: *Speedup* und Effizienz des gekoppelten Ansatzes bei Verwendung iterativer Gleichungslöser.

Wie bereits in Abschnitt 4.2 angesprochen, ist es möglich, unterschiedliche Gleichungslöser für die Berechnung zu wählen. Eine ausführliche Zusammenstellung und Diskussion unterschiedlicher iterativer Gleichungslöser ist in [135] zu finden. Die Bilder 7.6a und 7.6b zeigen den parallelen *Speedup* bzw. die parallele Effizienz des gekoppelten Verfahrens. In diesem konkre-

ten Fall sind die Ergebnisse zweier unterschiedlicher Gleichungslöser *conjugate gradient* - *cg* und *biconjugate gradient squared* - *bcgs* mit je zwei unterschiedlichen Vorkonditionierern *ilu* bzw. *icc* dargestellt. Sowohl Gleichungslöser als auch Vorkonditionierer werden von der Bibliothek PETSc bereitgestellt, nähere Informationen zur Effizienz der Gleichungslöser sind in [9] zu finden. Alle vier Konfigurationen liefern hier bei gleicher Rechengenauigkeit in etwa die gleichen Ergebnisse hinsichtlich *Speedup* und Effizienz. Bei Verwendung von bis zu zehn Prozessen liegt die parallele Effizienz bei 82% bis 95%, danach fällt sie merklich ab, bis sie bei 30 Prozessen einen Wert von etwa 50% erreicht.

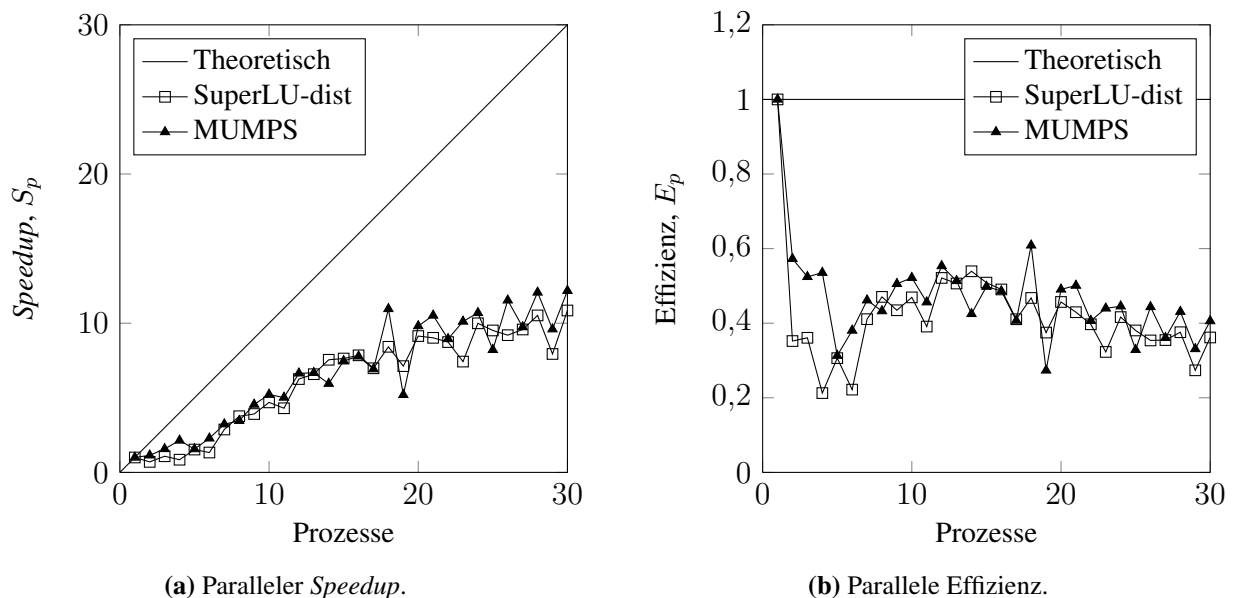


Bild 7.7.: ELPASO: *Speedup* und Effizienz des gekoppelten Ansatzes bei Verwendung direkter Gleichungslöser.

Alternativ können direkte Gleichungslöser eingesetzt werden. Dabei ist zu beachten, dass PETSc keine direkten Verfahren zum parallelen Lösen bereitstellt und auf andere Bibliotheken oder eigene Methoden zurückgegriffen werden muss. Die parallele Leistung des Programms bei Nutzung der direkten Löser ist in den Bildern 7.7a und 7.7b dargestellt. Die hier eingesetzten Gleichungslöser werden von den zuvor bereits erwähnten Bibliotheken MUMPS bzw. SuperLU bereitgestellt. Es ist deutlich zu sehen, dass die Kurve des *Speedup* im Vergleich zu den iterativen Lösern wesentlich flacher ausfällt. Die parallele Effizienz bei Verwendung von SuperLU liegt im Durchschnitt bei 41,9% im Falle von MUMPS bei 46,8%. Diese Zahlen lassen jedoch keine direkten Rückschlüsse auf die Gleichungslöser selbst zu, da hier das gesamte Lösungsverfahren betrachtet wird. Es ist dennoch offensichtlich, dass die iterativen Gleichungslöser bei gleicher Genauigkeit hier hinsichtlich *Speedup* und Effizienz zu besseren Ergebnissen führen.

Speedup und Effizienz beschreiben die Skalierung des Algorithmus bei einem gegebenen Problem, geben jedoch keine Informationen über die benötigte Rechenzeit an sich. In Bild 7.8 sind die benötigten Rechenzeiten normiert und logarithmisch dargestellt. Die Normierung erfolgt

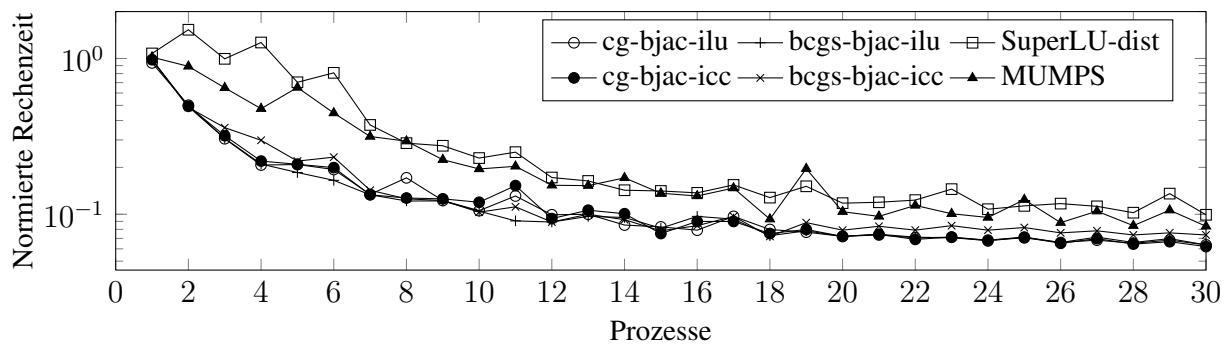


Bild 7.8.: Benötigte Rechenzeit in Abhängigkeit der gewählten Gleichungslöser.

über die durchschnittliche Rechenzeit aller Gleichungslöser auf einem Prozess, wobei die Abweichungen der Rechenzeit hier sehr gering ausfallen und somit die Wahl des Gleichungslösers nicht wesentlich ins Gewicht fällt. Im Weiteren, wenn die Anzahl der Prozesse steigt, sind die Unterschiede in der Rechenzeit deutlich zu erkennen. Dabei ist auffällig, dass es große Unterschiede bezüglich des gewählten Gleichungslösers gibt, wobei innerhalb der Gruppe der iterativen wie auch der direkten Gleichungslöser keine allzu großen Unterschiede auftreten. Mit den gewählten iterativen Gleichungslösern ist es möglich, die benötigte Rechenzeit für das beschriebene Beispiel bei einem Einsatz von zwölf Prozessen um 90% der sequentiellen Rechenzeit zu reduzieren. Im Vergleich dazu reduzieren die direkten Gleichungslöser die Rechenzeit lediglich um 85%. Diese Beobachtung lässt keine Rückschlüsse auf die Effizienz der Gleichungslöser selbst zu, sondern zeigt lediglich deren Effizienz innerhalb des gewählten Programmsystems.

7.4. Abschätzung des sequentiellen Programmanteils

Die genaue Bestimmung des sequentiellen Programmanteils ist sehr aufwändig und erfordert genaue Zeitmessungen und Untersuchungen der einzelnen Programmteile. Aus diesem Grund wird der gesamte sequentielle Programmanteil untersucht, um seine Größenordnung näherungsweise zu bestimmen. Dafür werden die Ergebnisse der Zeitmessung der starken Skalierung für SCABO und ELPASO mit den Überlegungen von AMDAHL verglichen. Durch Auswerten der Gleichung (7.3) wird jener Bereich bestimmt, der die gemessenen Ergebnisse einhüllt, sodass eine obere und eine untere Grenze des sequentiellen Anteils abgeschätzt werden können.

Bild 7.9a zeigt die zusammengefassten Messdaten des *Speedups* aus Bild 7.3a der starken Skalierung im Falle der Bestimmung des Fernfeldes mit SCABO. Grau hinterlegt ist der Bereich, in dem die Messdaten zu finden sein müssen. Es wird ein sequentieller Anteil von 2,5% bis 5% angenommen. Analog dazu zeigt Bild 7.9b die zusammengefassten Skalierungsergebnisse aus Bild 7.6a für die Berechnung des gekoppelten Ansatzes mit ELPASO. Auch hier ist der Bereich grau hinterlegt, der einem sequentiellen Programmanteil von 2,5% bis 5% entspricht. In beiden Fällen liegt ein Großteil der Messergebnisse innerhalb der grau eingefärbten Fläche, weshalb der sequentielle Anteil α der Berechnung sowohl für SCABO als auch ELPASO grob

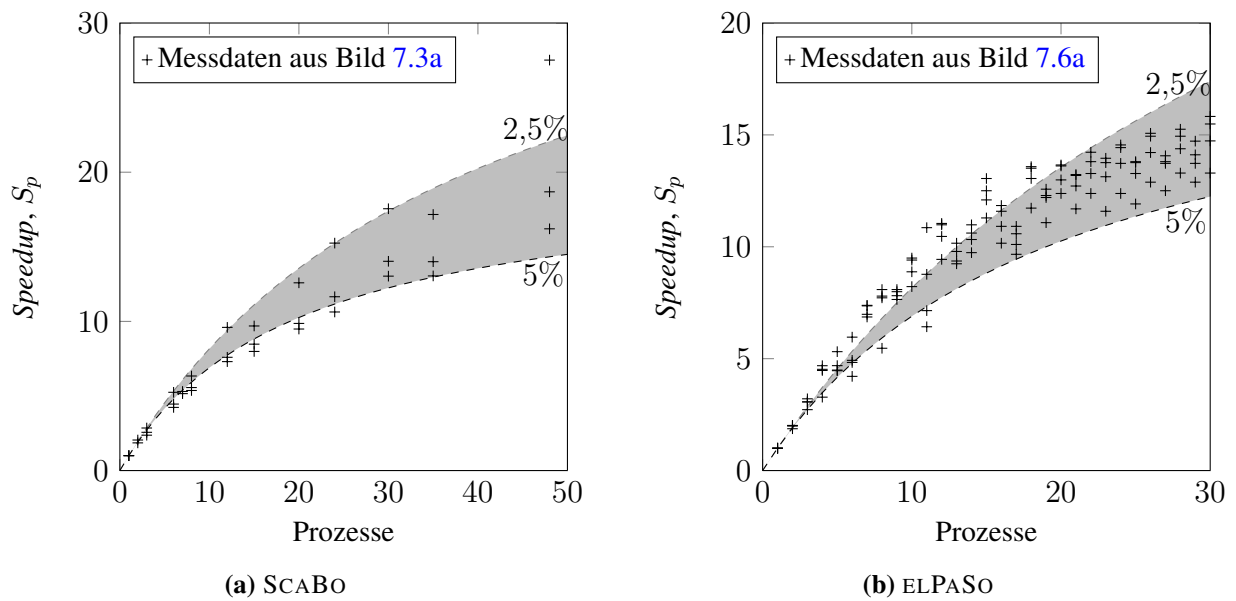


Bild 7.9.: Abschätzung des sequentiellen Programmanteils α für SCABO sowie für den gekoppelten Ansatz aus FEM und SBFEM in ELPASO. Grau hinterlegt ist der Bereich, der einem sequentiellen Anteil von $2,5\% \leq \alpha \leq 5\%$ entspricht.

mit 2,5% bis 5% abgeschätzt werden kann. Dies bedeutet gleichzeitig, dass der parallele Anteil der Software bei 95% bis 97,5% liegt. Ausgehend von diesen Zahlen ist ein theoretisch maximal erreichbarer *Speedup* von 20 bis 40 für die hier diskutierten Beispiele anzunehmen.

Abschließend ist festzuhalten, dass beide Programme gut skalieren und somit durch Hinzufügen weiterer Prozesse nicht nur zusätzlicher Arbeitsspeicher bereitgestellt, sondern auch die Rechenzeit signifikant reduziert werden kann.

8. Anwendungen in der Geotechnik

Die in dieser Arbeit beschriebene effiziente Kopplung von FEM und SBFEM wird durch die entwickelten Programme SCABO und ELPASO ermöglicht. Mit dieser Kopplung können in der aktuellen Version unterschiedliche geotechnische Fragestellungen untersucht werden. Da es sich hier um eine grundsätzliche Untersuchung des gekoppelten Ansatzes handelt, ist davon auszugehen, dass lediglich kleine Verformungen zu erwarten sind und lineare Materialmodelle zur Beschreibung von Struktur und Baugrund ausreichen. Unter diesen Voraussetzungen können beispielsweise konstante oder zeitveränderliche Einwirkungen auf Fundamente, Impulslasten auf Bauteile oder auch über den Baugrund auf benachbarte Gebäude eingeleitete Einwirkungen hinsichtlich der Verschiebungen oder Spannungen untersucht werden. Zusätzlich ist es möglich, mit der Software das dynamische Verhalten von Strukturen (Häuser, Brücken, Türme, Dammanlagen etc.) aufgrund von dynamisch einwirkenden Lasten in ihrer „natürlichen Umgebung“ zu untersuchen, da das gekoppelte Verfahren die Sommerfeld'sche Abstrahlbedingung erfüllt und somit eine Berücksichtigung des unendlichen Halbraumes gegeben ist. Damit ist im Gegensatz zur herkömmlichen FEM-Simulation eine Wellenreflektion am Diskretisierungsrand ausgeschlossen. Zu den dynamischen Lasten zählen beispielsweise vorbeifahrende Züge oder Lastkraftwagen, die aufgrund ihrer Vorbeifahrt Erschütterungen in den Boden einbringen. Da derartige Simulationen in der Regel mit einem hohen Berechnungsaufwand sowie Speicherbedarf einhergehen, wird die Software parallelisiert, um sie heutigen Multiprozessor-Architekturen als auch auf parallelgeschalteten und vernetzten Computern, wie beschrieben, effizient einsetzen zu können.

Für die Berechnung von Setzungen aufgrund konstanter Lasten gibt es für einfache Fundamentgeometrien und Lastverläufe analytische Lösungen, die zur Berechnung herangezogen werden können. Diese existieren jedoch nur für Fundamentgeometrien wie Rechteck oder Zylinder, auf die entweder Punkt-, Linien- oder Flächenlasten einwirken. Mit Hilfe einer numerischen Simulation können hingegen auch sehr komplexe Geometrien und Lastfälle untersucht werden. Der Einsatz des gekoppelten FEM-SBFEM-Ansatzes hat im Vergleich zu einer reinen FEM-Simulation in diesem Fall keinen Vorteil, wie in den nachfolgenden Untersuchungen zu sehen ist. Der Vorteil wird erst bei der Untersuchung von zeitveränderlichen Lasten erkennbar.

Werden dynamische Anregungen betrachtet, ist ein ausschließlich auf der FEM basierender Ansatz oftmals nicht ausreichend. Bei der Untersuchung derartiger Einwirkungen kommt der Vorteil des gekoppelten Ansatzes aufgrund dessen Fähigkeit, den offenen Rand ebenfalls zu berücksichtigen, zum Tragen. Ausgehend von einem Kontrollknoten im Berechnungsgebiet sind die Ergebnisse der FEM und FEM-SBFEM Simulation so lange deckungsgleich, bis die vom FEM-Gebietsrand reflektierte Welle an dem Kontrollknoten eintrifft. Danach unterscheiden sich die Ergebnisse deutlich voneinander und sind bei der FEM-Berechnung wegen der auftretenden Reflexion verfälscht. Somit muss die Gebietsgröße an das betrachtete Zeitfenster bei der FEM-Simulation angepasst werden, um diesen Effekt zu umgehen beziehungsweise zeitlich so weit zu verschieben, dass er nicht mehr betrachtet wird. Bei der gekoppelten Simulation ist dies

nicht nötig, da die Welle, wie beschrieben, das Gebiet verlassen kann und somit gar nicht erst reflektiert wird.

Zeitveränderliche Lasten können mit diesem Modell auch für numerische Integritätstests eingesetzt werden, um beispielsweise Fehlstellen in Pfählen zu simulieren und die Signale komplexer Pfahlgeometrien oder anderer Strukturen zu untersuchen. Dabei wird an entsprechender Stelle ein Impuls eingebracht und die Antwort des Systems über die Zeit aufgezeichnet. Da es zu keinen Reflexionen am Gebietsrand kommen kann, sind alle aufgezeichneten Antworten auf Geometrie- oder Materialänderungen zurückzuführen. Je nach untersuchter Geometrie und Bodenschichtung sind diese aufgezeichneten Signale beliebig komplex, wodurch eine einfache und eindeutige Zuordnung oftmals nicht möglich ist.

Auf der Grundlage einer numerischen Simulation lässt sich die Komplexität eines Modells schrittweise steigern und somit ein größeres Verständnis für die aufgezeichneten Signale erlangen. Bei Pfahl-Integritätstests mit der *Impact-Echo* Methode wird am Pfahlkopf ein Impuls eingebracht und die zeitliche Antwort des Systems aufgezeichnet und ausgewertet. Ziel ist es, so die Qualität des zu prüfenden Pfahls zu ermitteln und, wenn vorhanden, mögliche Fehler wie z. B. Querschnittsänderungen und deren Lage zu detektieren.

Ein weiterer Anwendungsfall ist die Untersuchung von zeitveränderlichen Lasten aufgrund von fest-installierten Maschinen, Verkehr oder Erdbeben und deren Auswirkung auf Bauwerke unterschiedlicher Art.

Nachfolgend werden drei unterschiedliche Anwendungsbeispiele diskutiert. Begonnen wird in Abschnitt 8.1 mit Fundamenten, die einer statischen oder dynamischen Belastung unterliegen. Daran anschließend werden in Abschnitt 8.2 numerische Pfahl-Integritätstests zur Fehlerdetektion durchgeführt. In dem letzten in dieser Arbeit diskutierten Beispiel soll ein kurzer Ausblick auf das, was mit der entwickelten Software in Zukunft untersucht werden könnte, gegeben werden. Dabei wird in Abschnitt 8.3 eine sehr stark abstrahierte Vorbeifahrt eines Zuges an Gebäuden mit unterschiedlicher Höhe und Ausdehnung betrachtet.

Unter Berücksichtigung höherwertiger Stoffmodelle im Nahfeld, die beispielsweise nicht-linear elastisches oder elastisch-plastisches Materialverhalten abbilden, können sich weitere Einsatzmöglichkeiten ergeben. Zu diesen zählen z. B. die Simulationen von Baugruben oder Böschungen um eventuelle Rutschungen vorhersagen zu können. In diesem Fall ist auch die Implementierung einer automatischen Neuvernetzung oder adaptiven Netzverfeinerung in Erwägung zu ziehen, damit große Verformungen und damit verbundene Spannungen korrekt dargestellt werden können. Werden zusätzlich Algorithmen zur Analyse von Kontaktproblemen ergänzt, könnten beispielsweise auch vollständige Rammprozesse von Pfählen oder Spundwänden numerisch simuliert werden. Bei derartigen Simulationen ist sicherzustellen, dass nicht-lineare Effekte nur innerhalb des Nahfeldes auftreten dürfen, da andernfalls das Fernfeld ebenfalls um höherwertige Stoffmodelle zu ergänzen ist. Dasselbe gilt im Falle der Neuvernetzung, auch hier muss das Fernfeld entsprechend angepasst werden, sobald Elemente auf dem gemeinsamen Kopplungsrand betroffen sind.

8.1. Statische und dynamische Belastung eines Fundaments

Die Verschiebungen und Spannungen unterhalb eines Fundaments werden nachfolgend betrachtet. Bild 8.1 skizziert das hier verwendete Nahfeld-Modell des im Boden eingebetteten Fundaments inklusive des berücksichtigten umgebenden Bodens. Der innere Quader bildet das Fundament mit einer Kantenlänge von $1,0 \text{ m} \times 1,0 \text{ m} \times 0,5 \text{ m}$. Dieser Quader ist von einem weiteren Quader der Kantenlänge $5,0 \text{ m} \times 5,0 \text{ m} \times 2,5 \text{ m}$ umgeben. Abschließend umfasst eine Halbkugel mit einem Radius von $10,0 \text{ m}$ die beiden Quader. Der zweite Quader sowie die Halbkugel bilden den umgebenden Boden ab. Für die spätere Vernetzung wird der zweite Quader in fünf Pyramidenstümpfe zerlegt. Die Halbkugel wird abzüglich der Pyramidenstümpfe ebenfalls zerlegt. Daraus folgen acht zusätzliche Volumenkörper für die Vernetzung (vgl. Bild 8.2).

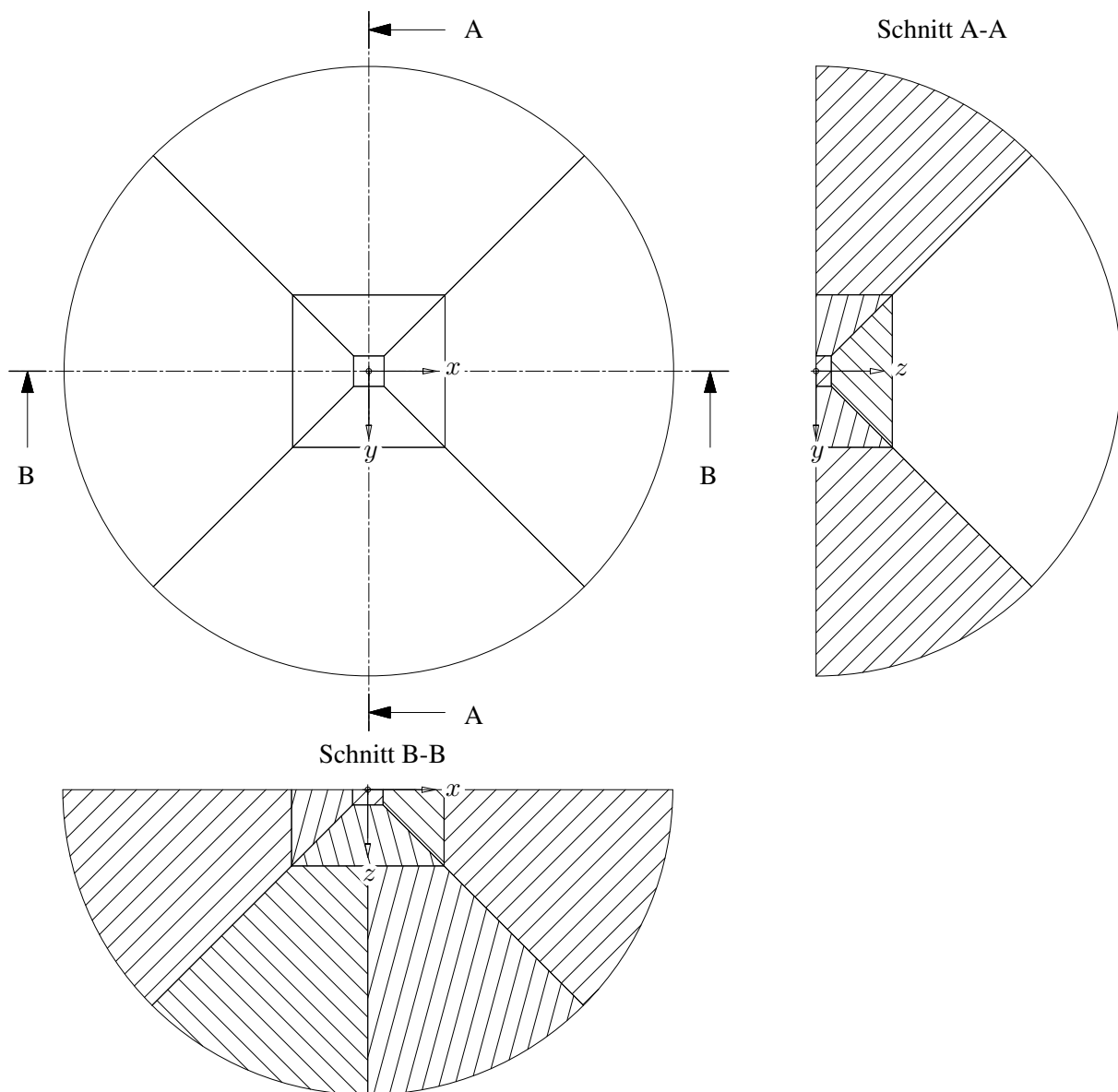


Bild 8.1.: Draufsicht sowie Schnitte des Nahfeldes.

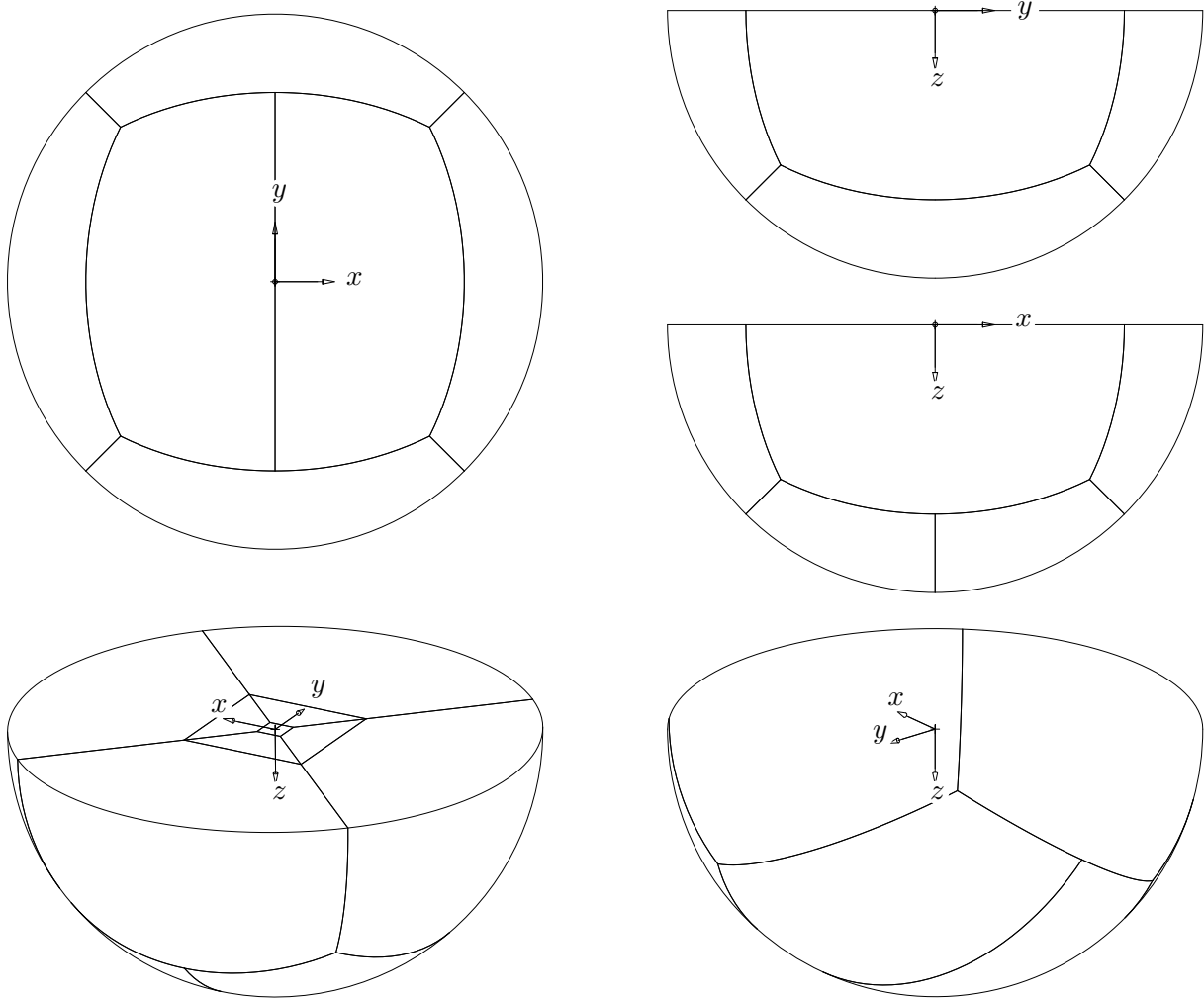


Bild 8.2.: Ansichten des Nahfeldes zweidimensional von unten, von zwei Seiten und dreidimensional von schräg oben sowie schräg unten.

Die Beschreibung der Vernetzung des in den Bildern 8.1 und 8.2 dargestellten Modells erfolgt von innen nach außen. Alle Volumen werden mit linearen dreidimensionalen Kontinuumsselementen (*Brick8* [17]) vernetzt. Das Fundament ist mit 20×20 Elementen in der xy -Ebene sowie zehn Elementen in der z -Richtung diskretisiert. Daraus folgt eine Elementkantenlänge von $\ell_{el} = 0,05$ m. Um eine ordnungsgemäße Vernetzung der einzelnen Volumen sicherzustellen wird gefordert, dass die Anzahl der Elemente an gemeinsamen Kanten einander entspricht. Dies gilt sowohl für die Anzahl der Kanten als auch für deren geometrische Lage. Damit das Seitenkantenverhältnis Höhe zu Breite zu Tiefe der Elemente relativ gleich ist und Elemente nicht unnötig verzerrt werden, wird der Knotenabstand in radialer Richtung derart angepasst, dass jedes Element möglichst gleichlange Kantenlängen aufweist.

Nach der Vernetzung besteht das Nahfeld aus 41200 Kontinuumsselementen mit 43322 Knoten. Die Diskretisierung des umgebenden Fernfeldes erfolgt mit 1200 Elementen (*Quad4* [142]) und 1241 Knoten unter Verwendung der SBFEM. Jeder Knoten im Nah- sowie Fernfeld verfügt, wie

schon in den vorangegangenen Validierungsbeispielen in Kapitel 5 beschrieben, über drei Freiheitsgrade, um die Verschiebungen im Raum vollständig abbilden zu können. Zur Reduktion der Rechenzeit wird die im Abschnitt 6.3 beschriebene Substrukturtechnik eingesetzt. Dafür wird das Fernfeld in vier Substrukturen entlang der xz - bzw. yz -Ebene zu je 300 Elementen mit je 331 Knoten zerlegt, sodass in diesem Fall alle Substrukturen die gleiche Größe aufweisen.

Neben dem gekoppelten Modell wird an dieser Stelle ebenfalls ein Modell unter ausschließlicher Verwendung der FEM betrachtet, sodass das Fernfeld entfällt und lediglich das Nahfeld untersucht werden kann. Da aufgrund dieser Tatsache der Einfluss des Fernfeldes entfällt, sind passende Randbedingungen zur Lagerung des Nahfeldes erforderlich. Sichergestellt wird die Lagerung des Nahfeldes, indem der äußere Rand in der xy -Ebene bei $z = 0$ in allen drei Raumrichtungen festgehalten wird. Es kann davon ausgegangen werden, dass die Verschiebungen dort ausreichend klein sind und die ins System eingebrachten Randbedingungen keinen wesentlichen Einfluss auf die Lösung haben. Diese Annahme wird im Weiteren durch den Vergleich mit der analytischen Lösung sowie dem gekoppelten Ansatz bestätigt. Alle anderen Knoten, die sich auf der Halbkugel befinden, werden zweiwertig in x - und y -Richtung festgehalten.

8.1.1. Verschiebungen und Spannungen unter Fundamenten

Die Verschiebungen aufgrund der konstanten Flächenlast eines ungestörten, homogenen, isotropen und idealelastischen Halbraums lassen sich, wie bereits in Abschnitt 5.2.1 erläutert, analytisch bestimmen. Die Auswertung erfolgt hier in Abhängigkeit der z -Achse, weshalb Gleichung (5.13) auszuwerten ist und nicht die vereinfachende Lösung nach Gleichung (5.18) herangezogen werden kann. Zusätzlich werden hier die Spannungen σ_z in Richtung der Lastaufprägung genauer betrachtet. Eine analytische Lösung ist in [111] oder [125] zu finden:

$$\sigma_z(z) = \frac{q}{2\pi} \left(\arctan \left(\frac{\ell b}{z \sqrt{\ell^2 + b^2 + z^2}} \right) + \frac{\ell b z}{\sqrt{\ell^2 + b^2 + z^2}} \left(\frac{1}{\ell^2 + z^2} + \frac{1}{b^2 + z^2} \right) \right). \quad (8.1)$$

Die Fläche der aufgeprägten Flächenlast $q = 40 \text{ kNm}^{-2}$ ist durch Länge $\ell = 1,0 \text{ m}$ und Breite $b = 1,0 \text{ m}$ des Fundaments gegeben. Wie schon bei der vorangegangenen Bestimmung der Setzung ist auch in diesem Fall die Spannung für σ_z an der Stelle $z = 0$ nicht definiert, kann jedoch für $\lim_{z \rightarrow 0} \sigma_z(z)$ asymptotisch angenähert werden. Erfolgt die Grenzwertbetrachtung für $\lim_{z \rightarrow \infty} \sigma_z(z)$ wird die Spannung zu Null.

Dem Boden sowie dem Fundament werden die Materialparameter für dicht gelagerten Sand mit den Kenngrößen $E = 100 \text{ MNm}^{-2}$, $\nu = 0,3$ und $\rho = 2200 \text{ kgm}^{-3}$ zugeordnet, um zu gewährleisten, dass die Gleichungen (5.13) und (8.1) gültig sind und zum Vergleich mit den numerischen Lösungen herangezogen werden können.

Die für die Simulation benötigte Zeitschrittlänge $\Delta t = \frac{\ell_{\text{el}}}{6c_p} \approx 3,0 \cdot 10^{-5} \text{ s}$ wird mit der kürzesten Elementlänge $\ell_{\text{el}} = 0,05 \text{ m}$ im Nahfeld sowie der Kompressionswellengeschwindigkeit $c_p = 247,36 \text{ ms}^{-1}$ bestimmt. Nach 200 Zeitschritten erfolgt die Extrapolation der Einflussmatrizen (vgl. Abschnitt 6.1).

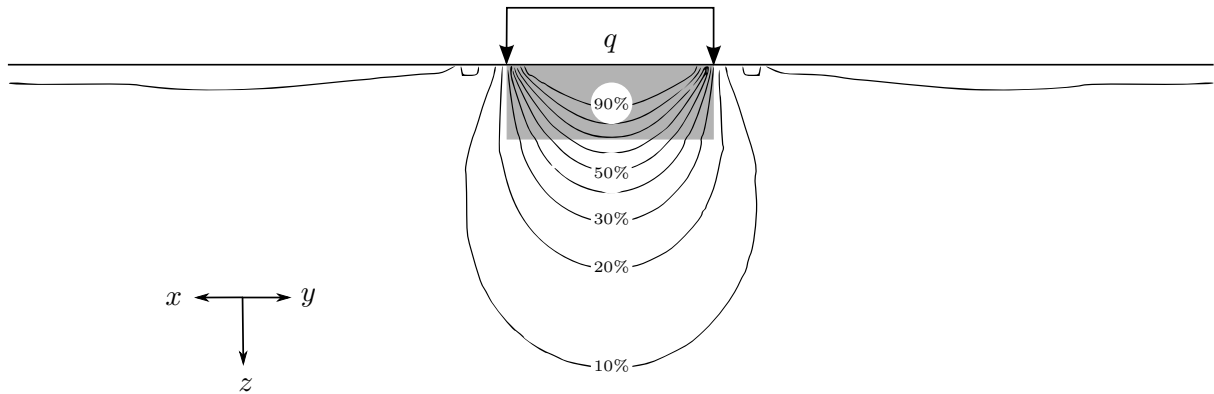
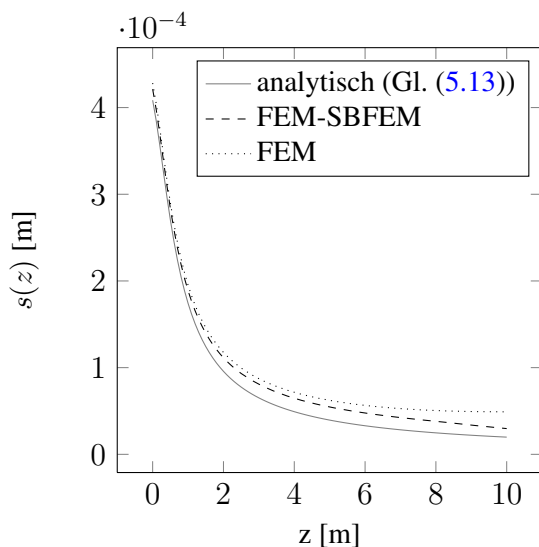
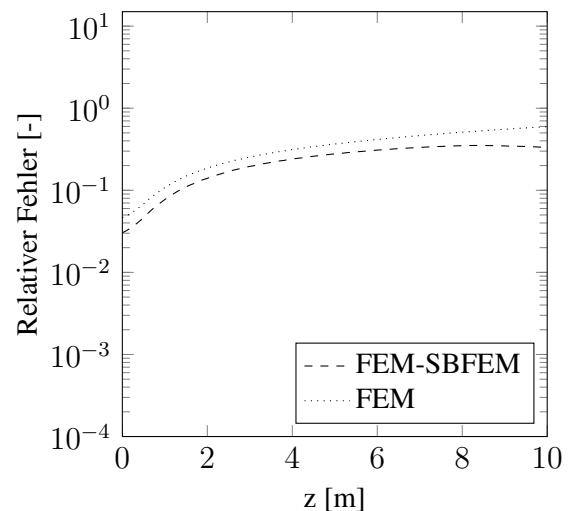


Bild 8.3.: Mit ELPASO berechneter Spannungsverlauf unterhalb des schlaffen Fundaments.

Unterhalb der eingeleiteten Flächenlast bildet sich die typische Druckzwiebel eines schlaffen Fundaments. Der Spannungsverlauf unterhalb der Flächenlast ist im Diagonalschnitt durch den Boden in Bild 8.3 dargestellt. Die analytische Lösung der Setzungsberechnung (vgl. Gl. (5.13)) liefert für den Mittelpunkt der Flächenlast einen Wert von $s(z \rightarrow 0) = 4,085 \cdot 10^{-4}$ m, dieser Wert wird von der FEM und der gekoppelten Simulation um 4,62 % bzw. 3,04 % überschätzt.



(a) Setzungsverlauf $s(z)$.



(b) Relativer Fehler der berechneten Setzung.

Bild 8.4.: Analytische und numerische Lösungen des Setzungsverlaufs in der Mitte der Belastung entlang der z -Achse sowie zugehörige relative Fehler.

In Bild 8.4a ist die Setzung unterhalb der Last dargestellt. Zusätzlich zu der analytischen Lösung sind in den Bildern zwei numerische Lösungen dargestellt. Zum einen die Simulation auf Basis der FEM, in diesem Fall handelt es sich um die Bestimmung der statischen und somit zeitlich unabhängigen Lösung. Zum anderen ist die Lösung des gekoppelten Verfahrens aus FEM und SBFEM zu sehen, in diesem Fall ist die zeitveränderliche Lösung bestimmt und die Simulation solange durchgeführt, bis sich die stationäre und somit zeitlich invariante Lösung eingestellt hat. Der zugehörige relative Fehler der Setzungsberechnung ist in Bild 8.4b zu finden. Es ist

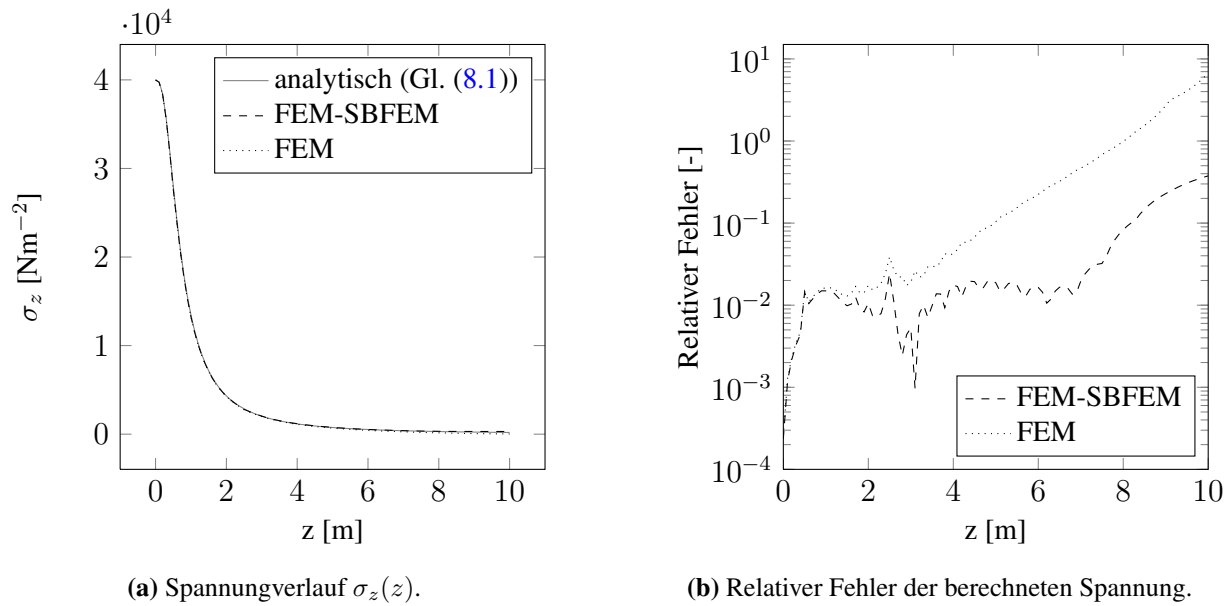


Bild 8.5.: Analytische und numerische Lösungen des Setzungs- und Spannungsverlaufs in der Mitte der Belastung entlang der z -Achse sowie zugehörige relative Fehler.

deutlich zu sehen, dass der gekoppelte Ansatz aus FEM und SBFEM näher an der analytischen Lösung liegt und somit das genauere Ergebnis liefert.

Die sich einstellende Spannung im Mittelpunkt der eingeleiteten Last ist in Bild 8.5a dargestellt. Die analytische Lösung wird anhand der beschriebenen Gleichung (8.1) bestimmt. Die beiden numerischen Lösungen nähern die analytische Lösung sehr gut an. Die Unterschiede in der Güte der Lösung werden erst bei der Betrachtung des relativen Fehlers

$$f = \left| \frac{\sigma_{z,\text{analytisch}} - \sigma_{z,\text{numerisch}}}{\sigma_{z,\text{numerisch}}} \right| = \left| \frac{\sigma_{z,\text{analytisch}}}{\sigma_{z,\text{numerisch}}} - 1 \right| \quad (8.2)$$

deutlich. In Bild 8.5b ist eindeutig zu erkennen, dass der relative Fehler mit zunehmendem Abstand zur Oberfläche anwächst. In einem Abstand von 10 m beträgt der relative Fehler der Spannung in z -Richtung im Fall der reinen FEM Rechnung etwa 6,8 %. Der relative Fehler des gekoppelten Modells beträgt hingegen lediglich 0,38 % und ist somit bei gleicher Auflösung der Diskretisierung eine Größenordnung genauer.

Wird für das Fundament ein deutlich steiferes Material (z. B. Beton B75, Festigkeitsklasse C60/75, $E = 39000 \text{ MNm}^{-2}$, $\nu = 0,15$ und $\rho = 2400 \text{ kgm}^{-3}$) als für den umgebenden Boden gewählt, entsteht eine gänzlich andere Spannungsverteilung, wie Bild 8.6 zeigt. Der dargestellte Spannungsverlauf unterscheidet sich stark von dem in der vorherigen Darstellung. Die Spannungen konzentrieren sich an den Rändern des Fundaments und sorgen somit für eine größere Beanspruchung des angrenzenden Bodens in diesem Bereich. Damit entspricht die numerische Simulation qualitativ auch den theoretischen bodenmechanischen Vorstellungen. Eine genaue quantitative Untersuchung der sich einstellenden Verschiebungen und Spannungen findet an dieser Stelle nicht statt.

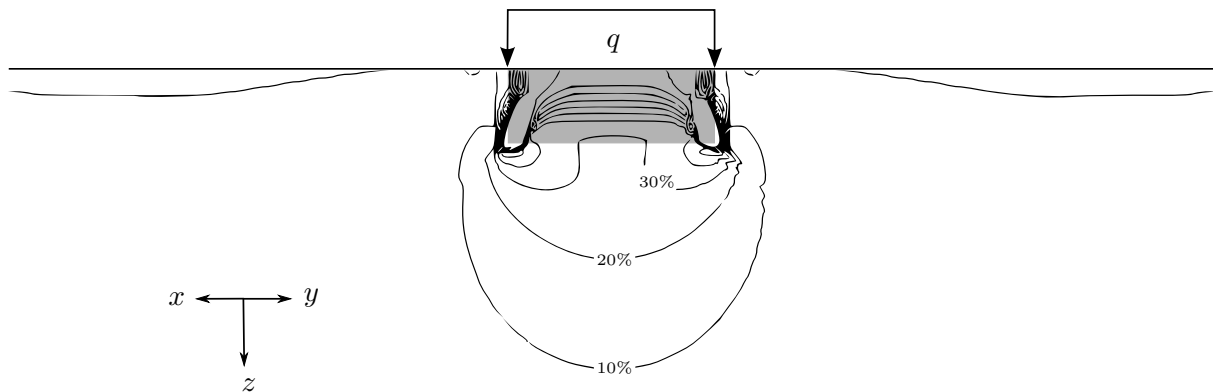


Bild 8.6.: Mit ELPASO berechneter Spannungsverlauf unterhalb des starren Fundaments.

8.1.2. Dynamische Belastung eines Fundaments

Neben der Auswirkung von statischen Lasten auf den Baugrund können auch dynamische, das heißt zeitveränderliche Lasten untersucht werden, wie sie beispielsweise bei Maschinenfundamenten auftreten. Dabei wird in der Regel das Fundament durch eine auf das Fundament montierte laufende Maschine dynamisch angeregt. In diesem Abschnitt wird zur Demonstration einer solchen dynamischen Anregung das im vorangegangenen Abschnitt beschriebene Fundament mit einer zeitlich veränderlichen Last mittig angeregt. Die gewählte Lastfunktion ist 2π -periodisch

$$\mathbf{f} = F_0 \begin{pmatrix} -\sin(2\pi\Omega t) \\ \cos(2\pi\Omega t) \\ \sin(2\pi\Omega t) \end{pmatrix}, \quad (8.3)$$

mit $F_0 = 1000 \text{ N}$ und $\Omega = 3,33 \text{ Hz}$. Das Fundament wird aufgrund der eingeleiteten Last in alle drei Raumrichtungen beansprucht. In Bild 8.7 ist die eingeleitete Last grafisch dargestellt.

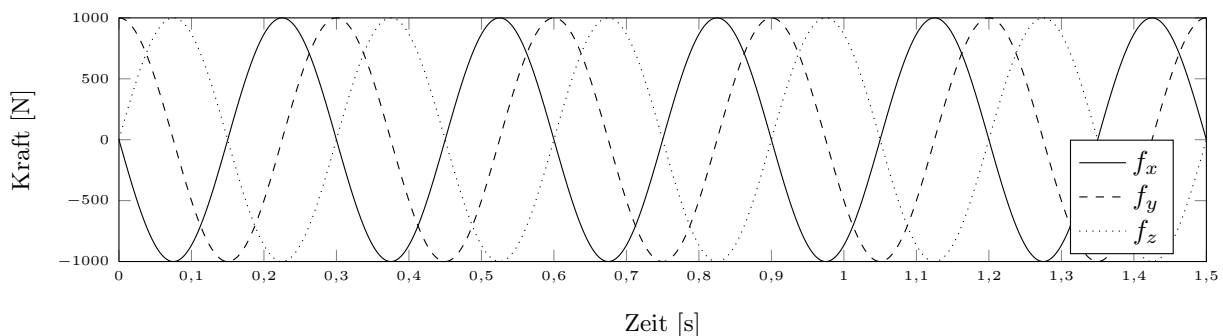


Bild 8.7.: Auf das Fundament aufgeprägte zeitlich veränderliche Last (vgl. Gleichung (8.3))

Die Geometrie sowie die Materialparameter entsprechen denen des starren Fundamentes aus dem vorherigen Beispiel, weshalb ebenfalls die Zeitschrittweite von $\Delta t = 3,0 \cdot 10^{-5} \text{ s}$ übernommen wird. Die Simulation der dynamischen Anregung des Fundaments erfolgt für fünf Perioden

und somit für die Zeitspanne von $T = 1,5$ s. Daraus folgen 50000 Zeitschritte, die berechnet werden müssen. Auch hier werden die Einflussmatrizen nach 200 Zeitschritten extrapoliert, so dass ein Verhältniswert der berechneten Einflussmatrizen des Fernfeldes zu Simulationsschritten des gekoppelten Systems, wie in Abschnitt 6.1 erläutert, von $\psi = \frac{t_m}{t_n} = \frac{200}{50000} = 0,004$ resultiert. Damit wird an dieser Stelle ebenfalls gezeigt, dass diese Methode auch für Simulationen mit größerer Anzahl an Zeitschritten unter Verwendung relativ weniger Einflussmatrizen stabil ist.

Ausgehend vom Lastangriffspunkt (Knoten 43322), der im Koordinatenursprung liegt, werden in x -Richtung zwei Referenzknoten im Abstand von 1,57 m (Knoten 26041), und 5,13 m (Knoten 9908) betrachtet. Die Verschiebung je Raumrichtung an diesen Knoten ist in Bild 8.8 sowohl für die gekoppelte FEM-SBFEM-Simulation als auch für die reine FE-Berechnung dargestellt.

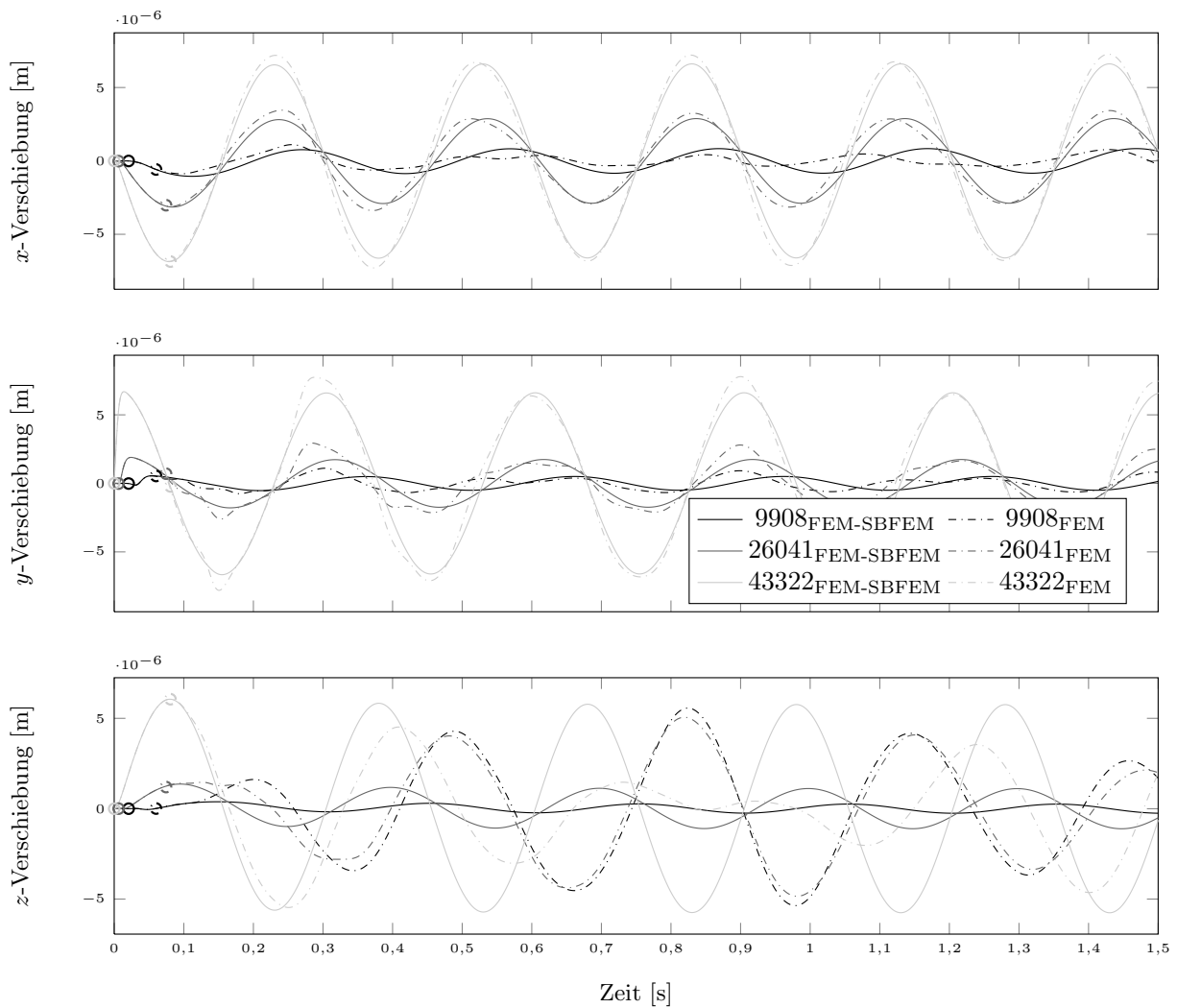


Bild 8.8.: Zeitabhängige Verschiebungen der Referenzknoten 43322, 26041 und 9908 je Raumrichtung.

Die mit dem gekoppelten Ansatz berechneten Verschiebungen der dargestellten Knoten geben den Verlauf der eingeleiteten Last wieder. Dabei entspricht die Verschiebung am Knoten 43322 genau dem Verlauf der Lastfunktion. Innerhalb der untersuchten 1,5 s zeigt die Verschiebung ebenfalls fünf Perioden. Aufgrund des Abstandes zum Lasteinleitungspunkt der zwei Referenzknoten setzt dort die Verschiebung zum einen zeitlich versetzt ein, zum anderen ist die Größe der Amplitude reduziert, da die eingeleitete Energie im Gebiet räumlich verteilt wird.

Die geschlossenen Kreise im Bild 8.8 markieren den Zeitpunkt, an dem die p-Welle die Knoten zum ersten Mal passiert. Vor diesem Zeitpunkt sind die Verschiebungen der Knoten gleich Null. Die offenen Kreise hingegen markieren den Zeitpunkt, an dem die Welle, die am Gebietsrand reflektiert wird, die dargestellten Knoten zum zweiten Mal passiert. Dies geschieht selbstverständlich nur im Falle der reinen Finite-Elemente-Berechnung. Bei der gekoppelten Simulation erfolgt diese Reflexion nicht. Bis zu diesem Zeitpunkt sind die Ergebnisse von FEM und FEM-SBFEM identisch. Danach unterscheiden sie sich erwartungsgemäß voneinander. Im Gegensatz zu dem vorangegangenen Beispiel, in dem der Unterschied zwischen FEM und FEM-SBFEM-Simulation aufgrund der statischen Last keinen großen Unterschied gemacht hat, ist hier der Vorteil des gekoppelten Verfahrens deutlich sichtbar. Um eine derartige Simulation ausschließlich mit der FEM für diesen Zeitraum durchzuführen, hätte das Nahfeld in seiner Ausdehnung deutlich größer ausfallen müssen, um sicherzustellen, dass die am Gebietsrand reflektierte Welle nicht innerhalb des betrachteten Zeitraumes wieder die beobachteten Punkte durchläuft.

8.2. Dynamische Belastung von Pfählen

Viele Ingenieurbauwerke wie Brücken, Hochhäuser, Türme oder Dämme und andere massive Bauwerke wie beispielsweise Kraftwerke oder Fabriken benötigen spezielle Fundamente, um die Lasten in den Baugrund abtragen zu können. Dabei kommen oftmals Pfahlgründungen aus vorgefertigten oder vor Ort hergestellten Pfählen zum Einsatz. Anhand zweier Beispiele wird nachfolgend gezeigt, dass es möglich ist, mit dem gekoppelten FEM-SBFEM Ansatz dynamisch belastete Pfähle im Baugrund numerisch zu untersuchen.

8.2.1. Pfahl-Integritätstests an Modellpfählen

Ein Anwendungsfall, der hier gezeigt werden soll, ist die numerische Simulation von Pfahl-Integritätstests (PIT). Je nach Art der Herstellung der Pfähle kann es zu unterschiedlichen Schädigungen während der Herstellung oder des Einbaus in den Baugrund kommen. Bei vorgefertigten Pfählen kann es durch den Rammprozess zur Schädigung des Pfahls kommen, sodass der Pfahl Risse bekommt oder absichert, falls er unbewehrt ausgeführt ist. Bei vor Ort hergestellten Pfählen kann es aufgrund des anstehenden Baugrunds zu Einschnürungen in der Pfahlgeometrie, aber auch zu Aufweitungen kommen, wenn sich der horizontale Erddruck signifikant vom hydrostatischen Druck des eingebrachten Betons unterscheidet. Auch nach dem Herstellungs-

prozess der Pfähle können sie durch Arbeiten auf der Baustelle und dadurch einhergehende horizontale Belastungen beschädigt werden. Eine detaillierte Auflistung der möglichen Schädigungen von Pfählen ist in [86] zu finden.

Die Schädigung von Pfählen oder besser deren Integrität kann im eingebauten Zustand (in situ) anhand eines Pfahl-Integritätstests geprüft werden. Der PIT ist ein zerstörungsfreies Testverfahren, das in der Literatur auch unter den Begriffen *Low-Strain-Methode* oder *impact echo testing* zu finden ist [52,57]. Dabei wird am Pfahlkopf ein Sensor installiert, der den ebenfalls am Pfahlkopf eingebrachten Impuls aufzeichnet. Anhand der Auswertung der aufgezeichneten Daten ist es möglich, die Länge des Pfahls zu prüfen und Fehlstellen zu lokalisieren. Des Weiteren lassen sich Aussagen bezüglich der Betonqualität treffen. Liegt die beobachtete Wellenausbreitungsgeschwindigkeit unterhalb der erwarteten Wellengeschwindigkeit, kann dies ein Indiz für eine geringere Betonqualität sein.

Bei der *Low-Strain-Methode* durchläuft ein mit einem Handhammer aufgebracht Impuls den zu untersuchenden Pfahl. Dabei setzt sich der Impuls mit der für den Pfahl charakteristischen Wellengeschwindigkeit c fort. Am Pfahlfuß wird ein Teil der Welle reflektiert und schließlich am Pfahlkopf von einem Sensor registriert und für die spätere Auswertung aufgezeichnet.

Die Laufzeit der Welle ist durch

$$T = \frac{2\ell}{c} \quad (8.4)$$

gegeben und entspricht somit genau der Zeit, die die Welle benötigt, um vom Pfahlkopf, wo sie durch den Impuls eingeleitet wird, bis zum Pfahlfuß und wieder zurück zum Pfahlkopf zu laufen. Sind die Pfahllänge und das Material bekannt, können Veränderungen in der Laufzeit direkt mit Veränderungen im Pfahl, wie beispielsweise Störstellen, in Zusammenhang gebracht werden.

Die am Pfahlkopf eingeleitete Kraft F ist proportional zur Pfahlkopfgeschwindigkeit v , für den Fall, dass von einem vereinfachenden eindimensionalen Ersatzmodell ausgegangen wird (vgl. Abschnitt 5.1.3). Der Proportionalitätsfaktor Z entspricht damit Gleichung (5.10), analog zu dem Validierungsbeispiel gilt auch hier Gleichung (5.9). In der vereinfachten Beschreibung wird die Wechselwirkung zwischen Pfahl und dem seitlich anstehenden Boden vernachlässigt. Der Reflexionsfaktor wird mittels der Impedanzen Z_1 und Z_2 des Impedanzwechsels bestimmt (Gleichung (5.11)). Dieser beschreibt den Anteil der Welle, der am Impedanzsprung reflektiert wird. Ein Impedanzwechsel tritt immer dann auf, wenn sich der Querschnitt A oder die Steifigkeit bzw. Dichte des Materials verändern. Der Transmissionsfaktor wird mit Gleichung (5.12) bestimmt. Er beschreibt den Anteil der Welle, der nach dem Impedanzsprung nicht reflektiert wird und somit seine Ausbreitungsrichtung beibehält. Bei einem fehlerfreien Pfahl erfolgt der erste Impedanzsprung am Pfahlfuß beim Übergang in den Baugrund. Ist der Pfahl jedoch nicht fehlerfrei, kommt es bei jeder Veränderung des Pfahlquerschnittes oder des Materials zu einem Impedanzsprung, der bei der Messung sichtbar wird und dem entsprechenden Fehler zugeordnet werden muss, damit er aus ingenieurtechnischer Sicht beurteilt werden kann.

8.2.1.1. Numerische Simulation

Modelliert ist ein Pfahl mit einer Länge $\ell_0 = 2,1$ m und einem konstanten Radius $r_0 = 0,1$ m. Der Pfahlfuß befindet sich 2,0 m unterhalb der Geländeoberfläche, die oberen 0,1 m des Pfahls schauen heraus. Dieser Pfahl gilt als Prototyp und wird als Referenzlösung für die fehlerbehafteten Pfähle herangezogen. Des Weiteren werden vier Pfähle mit definierten Fehlern untersucht. Bei diesen Pfählen ist der Radius abschnittsweise modifiziert und somit nicht über die gesamte Pfahllänge konstant. Es wird sowohl ein reduzierter Radius $r_1 = \frac{3}{4}r_0$ als auch ein erweiterter Radius $r_2 = \frac{5}{4}r_0$ betrachtet. Die Länge der Pfähle ist bei allen hier betrachteten Pfählen unverändert. Die zu untersuchenden Pfahlgeometrien sind in Tabelle 8.1 zusammengefasst.

z -Richtung [m]		Referenzpfahl	Fehler Typ 1	Fehler Typ 2	Fehler Typ 3	Fehler Typ 4
[+0. 1]	Pfahlkopf	r_0	r_0	r_0	r_0	r_0
[+0. 1; $\pm 0. 0$]		r_0	r_0	r_0	r_0	r_0
[$\pm 0. 0$]	Oberfläche	r_0	r_0	r_0	r_0	r_0
[$\pm 0. 0$; $-0. 5$]		r_0	r_0	r_0	r_0	r_0
[$-0. 5$; $-1. 0$]		r_0	r_0	r_0	r_0	r_2
[$-1. 0$; $-1. 5$]		r_0	r_0	r_1	r_2	r_2
[$-1. 5$; $-2. 0$]		r_0	r_1	r_1	r_2	r_2
[$-2. 0$]	Pfahlfuß	r_0	r_1	r_1	r_2	r_2

Tabelle 8.1.: Nicht maßstabsgetreue Darstellung der rotationssymmetrischen Pfähle. Fehlerfreier Referenzpfahl mit Radius r_0 sowie fehlerbehaftete Pfähle mit r_1 bzw. r_2 .

Für den Pfahl werden die folgenden Materialparameter E-Modul $E = 37000 \text{ MNm}^{-1}$, Poissonzahl $\nu = 0,2$ und Dichte $\rho = 2,6 \text{ tm}^{-3}$ gewählt. Für den umgebenden Baugrund werden die folgenden Werte angenommen: $E = 100 \text{ MNm}^{-2}$, $\nu = 0,3$ und $\rho = 2,1 \text{ tm}^{-3}$. In beiden Fällen handelt es sich um isotropes, homogenes und ideal-elastisches Material.

Auf Grundlage der Materialparameter werden die maximalen Wellengeschwindigkeiten für die zeitliche Diskretisierung bestimmt. Die p-Wellengeschwindigkeit im Pfahl beträgt $3976,43 \text{ ms}^{-1}$ für den Boden resultiert eine Geschwindigkeit von $253,18 \text{ ms}^{-1}$.

Die p-Welle benötigt $1,0562 \cdot 10^{-3} \text{ s}$, um den Pfahl vom Pfahlkopf zum Pfahlfuß und zurück zum Pfahlkopf zu durchlaufen. Die Dehnwelle mit einer Geschwindigkeit von $3772,37 \text{ ms}^{-1}$ benötigt für denselben Weg $1,1134 \cdot 10^{-3} \text{ s}$ (vgl. Gleichung (8.4)).

Bei der numerischen Simulation des Pfahl-Integritätstests werden im Nahfeld sowohl der Pfahl als auch ein Teil des umgebenden Bodens mit finiten Elementen modelliert. Dies ermöglicht im Weiteren eine Anpassung der Pfahlgeometrie im Nahfeld, ohne dabei gleichzeitig die Vernetzung des Fernfeldes ändern zu müssen. Zur Reduktion der Rechenzeit wird lediglich ein Viertel des Gesamtsystems betrachtet, wodurch in diesem Fall ausnahmslos rotationssymmetrische Fehler untersucht werden können. In der xz -Ebene werden die Freiheitsgrade in y -Richtung festgehalten und in yz -Richtung dementsprechend die Freiheitsgrade in x -Richtung.

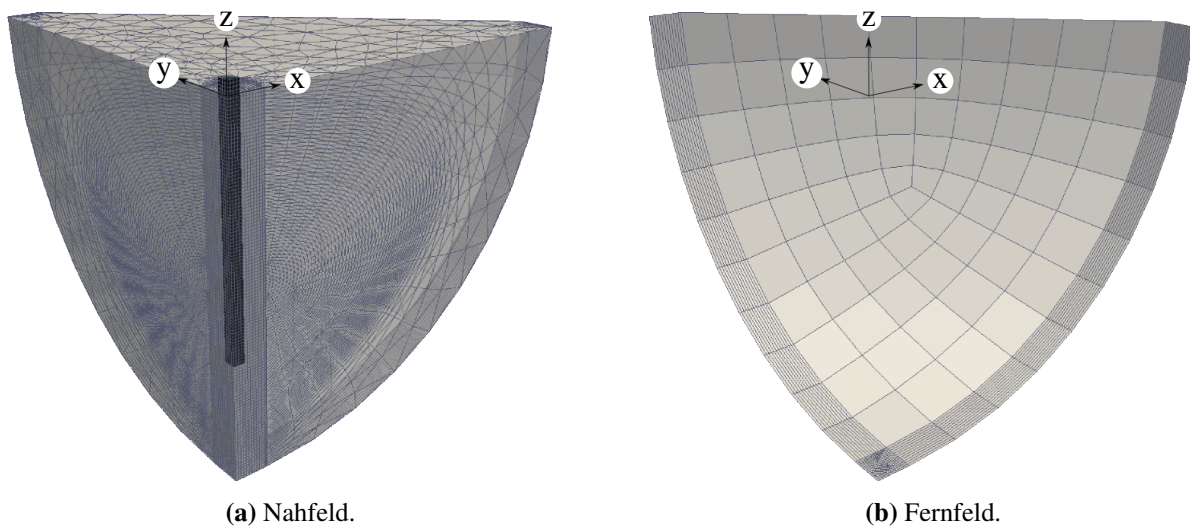


Bild 8.9.: Diskretisierung des Nahfeldes mit Pfahl und umgebendem Boden unter Verwendung finiter Elemente in (a) sowie Diskretisierung des Fernfeldes mit der *Scaled Boundary Finite Element Method* in (b).

Bild 8.9 zeigt das vernetzte Nahfeld. Im Inneren ist der zu untersuchende Pfahl zu sehen. Die Vernetzung des Pfahls sowie des Bodens in direkter Nähe zum Pfahl erfolgt wie schon bei den vorangegangenen Beispielen mit *Brick8* Elementen. Die Elemente haben eine maximale Kantenlänge von 0,025 m. Hinzu kommen weitere *Tetra4* Elemente, mit denen der vom Pfahl weiter entfernte Boden diskretisiert ist. Beide Elementtypen nutzen lineare Ansatzfunktionen [17]. Die Diskretisierung des umgebenden Fernfeldes erfolgt mit Elementen vom Typ *Quad4* [142]. Das Skalierungszentrum liegt im Ursprung des Koordinatensystems. Der Abstand vom Skalierungszentrum zum Rand beträgt zwischen 1,8 m und 3,3 m (vgl. Gleichung (8.4)).

	Knoten	Elemente	Freiheitsgrade
FEM	50611	28140 <i>Brick8</i>	151833
		116974 <i>Tetra4</i>	
SBFEM	589	549 <i>Quad4</i>	1794

Tabelle 8.2.: Pfahl-Integritäts-Tests Netzinformation.

Tabelle 8.2 fasst die für dieses Beispiel relevanten Netzinformationen zusammen. Die Simulation erfolgt mit 151833 Freiheitsgraden im Nahfeld und 1794 Freiheitsgraden im Fernfeld. Dieselbe Diskretisierung wird bereits in [138] untersucht. Es werden 2000 Zeitschritte mit einer Zeitschrittlänge von $6 \cdot 10^{-6}$ s berechnet. Die Zeitschrittintegration erfolgt mit dem Generalized- α -Verfahren unter Verwendung der folgenden Integrationsparameter $\alpha_m = 0,125$, $\alpha_f = 0,375$, $\beta = 0,390625$ und $\gamma = 0,75$.

Der Impuls, mit dem der Pfahl beaufschlagt wird, wird in diesem Anwendungsbeispiel vollflächig auf den Pfahlkopf aufgebracht. Es werden somit alle Knoten am Pfahlkopf gleichzeitig belastet und wieder entlastet. Dies geschieht ähnlich wie im Beispiel 1 (vgl. Abschnitt 5.1). Auch hier wird ebenfalls ein Ricker *Wavelet*

$$p = (-50 - q) e^{-q} \text{ N mit } q = (5t\pi - 6)^2 \quad (8.5)$$

als Lastfunktion genutzt.

8.2.1.2. Ergebnisse

Wird die numerische Simulation der Pfahl-Integritätstests für die in Abschnitt 8.2.1.1 beschriebenen Pfahlgeometrien durchgeführt und die sich einstellenden Beschleunigungen, Geschwindigkeiten sowie Verschiebungen am Pfahlkopf mitgeschrieben, folgen daraus die in den Bildern 8.10 und 8.11 dargestellten zeitlichen Verläufe.

Der eingeleitete Impuls ist in allen drei Darstellungen deutlich zu erkennen (vgl. Bild 8.10). Wie zu erwarten wird nach einer Periode T (vgl. Gleichungen (8.4)) die Antwort des am Pfahlfuß reflektierten Impulses am Pfahlkopf registriert und noch ein weiteres Mal nach $2T$. Es ist zu sehen, dass die Amplitude der Ausschläge nach jeder Reflexion deutlich kleiner wird, was darauf zurückzuführen ist, dass lediglich ein Teil der Welle am Pfahlfuß reflektiert wird und der andere Teil in den Baugrund übergeht, um dann im unendlichen Fernfeld dissipiert zu werden.

Wird der Reflexionsfaktor wie beschrieben bestimmt, folgt er zu ca. 91% und der Transmissionsfaktor entsprechend zu 9% (vgl. Gleichungen (5.11) und (5.12)). Das heißt, dass 91% der durch den Impuls verursachten Geschwindigkeit v_{\max} am Pfahlkopf zum Zeitpunkt T aufgezeichnet werden müssten. Zum Zeitpunkt $2T$ sollten immerhin noch 83% der Geschwindigkeit v_{\max} registriert werden. Dass dies nicht der Fall ist, ist leicht ersichtlich. Der Grund findet sich im Modell, das dem eingangs diskutierten eindimensionalen Ersatzmodell nicht entspricht. Zum einen ist der Pfahl in den Boden eingebettet, wodurch ein Teil der Welle seitlich durch den Mantel des modellierten Pfahls in den Boden übergeht. Zum anderen verliert die Welle an Energie aufgrund der im numerischen Modell berücksichtigten Dämpfung. Um aufgezeichnete Reflexionen trotz der Dämpfung besser sichtbar zu machen, ist es nötig, das mitgeschriebene Signal mit Hilfe einer Verstärkerfunktion zu bearbeiten [86].

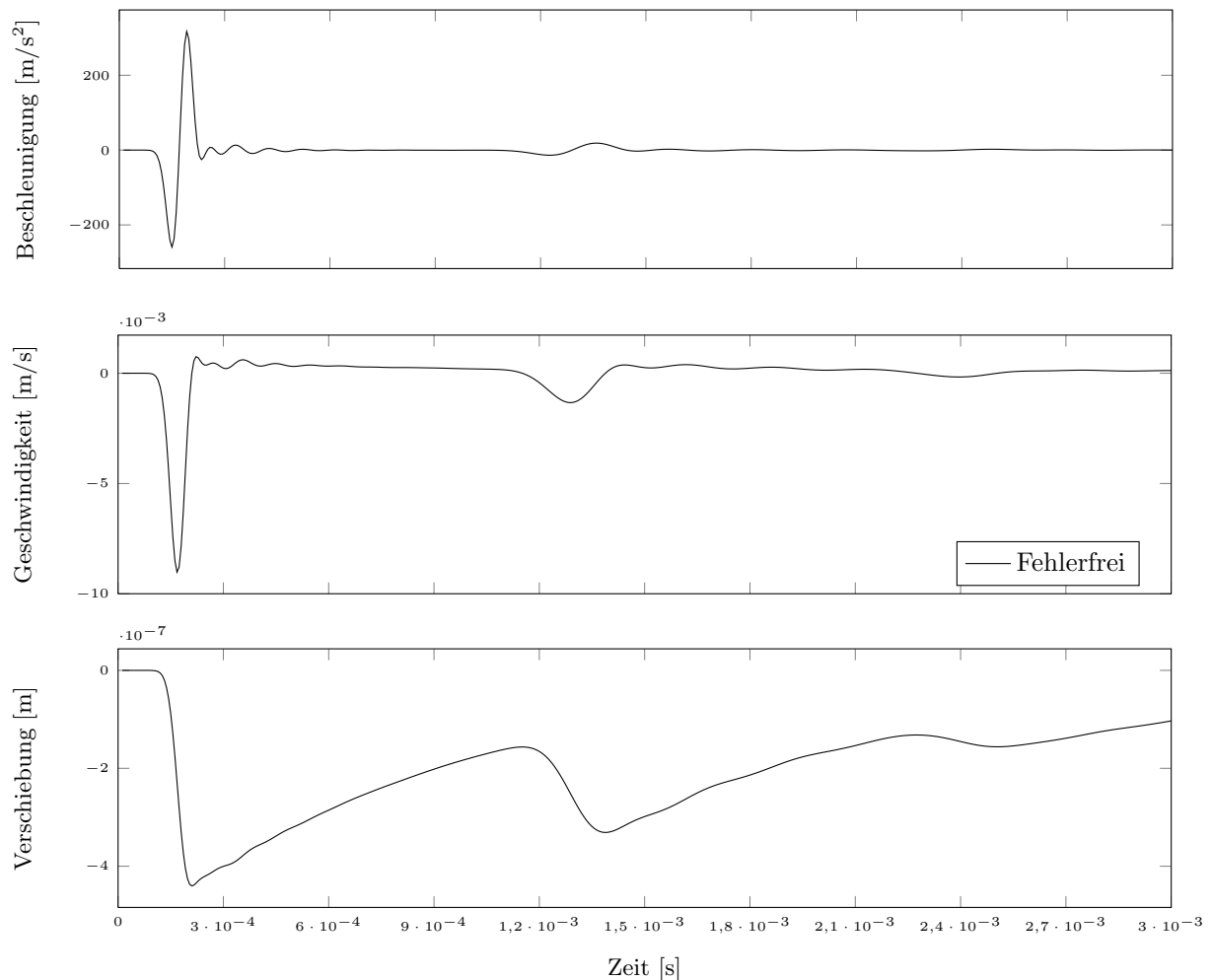


Bild 8.10.: Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf des Referenzpfahls in z -Richtung.

In Bild 8.11 sind die zeitlichen Verläufe der Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf der fehlerbehafteten Pfähle dargestellt. Zusätzlich sind zum direkten Vergleich die Verläufe des fehlerfreien Pfahls aus Bild 8.10 als Referenzlösung dargestellt.

Die größten Unterschiede in den zeitlichen Verläufen der einzelnen Pfahltests sind offensichtlich bei den Verschiebungen zu sehen. Zum Zeitpunkt T zeigen auch die Pfähle, bei denen der Radius fehlerhaft modelliert wird, einen Ausschlag im Diagramm. Dasselbe gilt ebenfalls für den Zeitpunkt $2T$. Es treten jedoch schon vorher Reflexionen auf ($\leq T$), die am Pfahlkopf registriert werden. Die vorzeitigen Reflexionen lassen sich den einzelnen eingprägten Fehlern eindeutig zuordnen.

Wird der Radius verringert, erfolgt durch die Reduktion des Pfahlquerschnittes eine vorzeitige und teilweise Reflexion der Welle, wodurch der untersuchte Pfahl kürzer erscheint, wie es bei den Fehlern 1 und 2 der Fall ist. Da die Reduktion des Pfahlquerschnittes im Fall 2 etwa in Pfahlmitte erfolgt, wird die teilweise Reflexion am Pfahlkopf früher als im Fall 1, bei dem die Verringerung des Radius erst im unteren Viertel des Pfahls erfolgt, registriert. Der

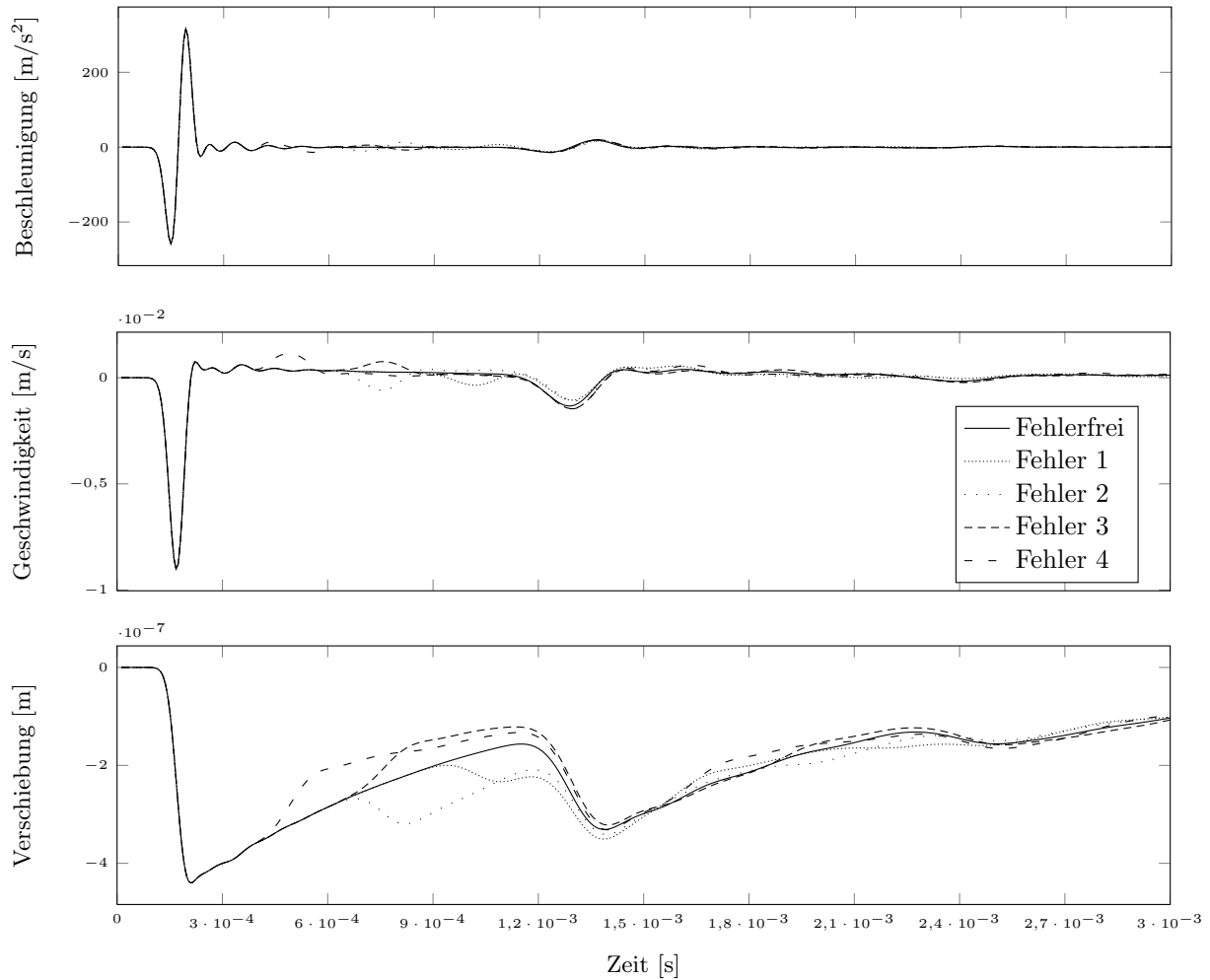


Bild 8.11.: Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf in z -Richtung.

exakte Zeitpunkt der Reflexion kann bestimmt werden, indem bei der Ermittlung der Periode $T' = \frac{\ell_0 - \ell_F}{\ell_0} T$ die Position ℓ_F der Querschnittsänderung berücksichtigt wird (vgl. Tabelle 8.1). Dabei ist ℓ_F genau der Abstand zwischen Fehlstelle und Pfahlfuß. Somit kann $T'_2 = \frac{2,1-1,0}{2,1} T$ bzw. $T'_1 = \frac{2,1-0,5}{2,1} T$ für beide Fälle bestimmt werden. Analog gilt dies ebenfalls für die untersuchten Fehler 3 und 4. Auch für diese beiden kann, anhand des Zeitpunktes der registrierten Reflexion, der Fehler eindeutig zugeordnet werden. Im Gegensatz zu den Fehlern 1 und 2 sehen die Verläufe deutlich anders aus. Grund dafür ist die Phasenumkehr aufgrund der Erweiterung des Querschnittes. Damit sind die hier vorgestellten Ergebnisse konsistent zu den in der Literatur zu findenden Beispielen [52, 124].

Diese und ähnliche Daten wurden in dem Projekt „DAAD-IKYDA 2012: *Improvement of integrity monitoring of piles using computational dynamics and global optimization*“, einer deutsch-griechischen Kooperation, genutzt, um mit Hilfe selbstlernender Algorithmen fehlerhafte Pfähle zu identifizieren und die numerisch generierten Daten den jeweils korrespondierenden Fehlern zuzuweisen [127, 140]. Dabei konnten die Fehler je nach Datengrundlage mit einer Genauigkeit von bis zu 100% zugeordnet werden.

8.2.2. In situ Rammtest

In Analogie zum Pfahl-Integritätstest soll eine Rammsimulation durchgeführt werden. Da die Software in der aktuellen Version weder große Verformungen noch eine Neuvernetzung vorsieht und der Pfahlvortrieb in den Boden somit nicht dargestellt werden kann, wird davon ausgegangen, dass die Einbindetiefe bereits erreicht ist und der Pfahl auf einer Felsschicht steht. Der Pfahl verändert somit trotz weiterer Rammschläge seine aktuelle Position nicht.

Ein Pfahl mit einer Länge von 21 Metern und einem Durchmesser von einem Meter wird 20 Meter in sandigen Boden gerammt und steht nun auf einer festen Schicht, deren E-Modul 80% des Pfahl E-Moduls entspricht. Somit befindet sich der Pfahlkopf einen Meter oberhalb des Bodens. Die Materialkennwerte des Pfahls sowie der beiden Bodenschichten sind in Tabelle 8.3 zusammengefasst.

	E-Modul MNm^{-2}	Poissonzahl	Dichte tm^{-3}
Pfahl	37000,0	0,2	2,6
Erste Bodenschicht	100,0	0,3	2,1
Zweite Bodenschicht	29600,0	0,3	2,6

Tabelle 8.3.: Zusammenfassung der Materialkenngrößen des Rammtests.

Bodenkonfigurationen 1	Knoten	Elemente	Freiheitsgrade
FEM	50611	28140 <i>Brick8</i> 116974 <i>Tetra4</i>	151833
SBFEM Sub 1	275	182 <i>Quad4</i>	825
SBFEM Sub 2	483	367 <i>Quad4</i>	1449

Bodenkonfigurationen 2	Knoten	Elemente	Freiheitsgrade
FEM	50611	28140 <i>Brick8</i> 116974 <i>Tetra4</i>	151833
SBFEM Sub 1	244	147 <i>Quad4</i>	732
SBFEM Sub 2	522	402 <i>Quad4</i>	1566

Tabelle 8.4.: Netzinformation des durchgeführten Rammtest für die beiden Bodenkonfiguration.

Die Simulation erfolgt mit zwei unterschiedlichen Diskretisierungen, die sich in der Modellierung der Bodenschichten unterscheiden. Die Anzahl der Knoten und damit auch der Elemente im Nahfeld ist in beiden Fällen gleich. Im ersten Modell werden alle Elemente, die sich vollständig unterhalb des Pfahlfußes befinden, der festen Schicht zugeordnet. Im zweiten Fall erfolgt diese Zuordnung für alle Elemente, die mindestens einen Knoten unterhalb des Pfahlfußes

ßen haben. Aufgrund der Vernetzung sind die beiden Bodenschichten nicht parallel zueinander, sondern miteinander verzahnt (vgl. Bild 8.9a).

Das Fernfeld wird für jede der beiden Bodenkonfigurationen in zwei Substrukturen zerlegt, sodass die Einflussmatrizen jeder Bodenschicht für sich berechnet werden. Eine Zusammenfassung bezüglich der Anzahl an Knoten und Elementen ist Tabelle 8.4 zu entnehmen. Erst bei der Simulation von Nahfeld und Fernfeld werden die Substrukturen gemeinsam an das Nahfeld gekoppelt, sodass das gesamte Fernfeld berücksichtigt wird. Die Simulation des Rammtests wird für $3 \cdot 10^{-2}$ s mit einer Zeitschrittlänge von $\Delta t = 6 \cdot 10^{-6}$ s durchgeführt. In der vorgeschalteten Berechnung des Fernfeldes werden 500 Einflussmatrizen bestimmt. Alle weiteren Einflussmatrizen des Fernfeldes für die benötigten 5000 Zeitschritte werden aufgrund der berechneten Daten extrapoliert.

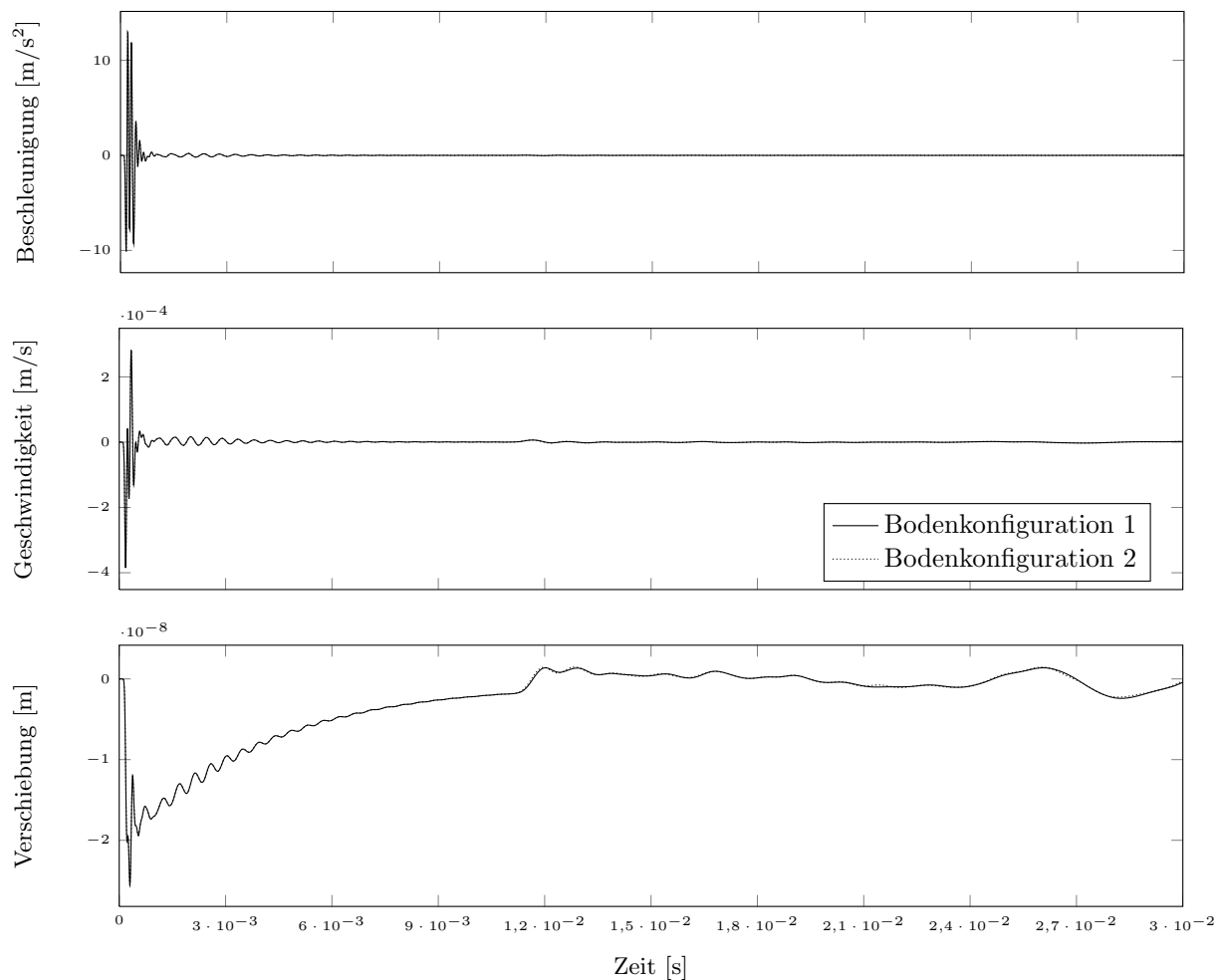


Bild 8.12.: Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf für die zwei unterschiedlichen Bodenkonfigurationen.

Bild 8.12 zeigt die resultierende Beschleunigung, Geschwindigkeit und Verschiebung in Pfahlkopfmittle für die beiden simulierten Bodenkonfigurationen. Aufgrund der unterschiedlichen Modellierung der beiden Bodenschichten unterscheiden sich die Ergebnisse der Bodenkonfigu-

rationen nur leicht voneinander. Sie lassen sich am ehesten in der Darstellung der Verschiebungen ausmachen. Im Falle der Bodenkonfiguration 2 erfolgt die Reflexion der Welle aufgrund der höher beginnenden festen Bodenschicht $6,63 \cdot 10^{-5}$ s früher und wird damit $1,33 \cdot 10^{-4}$ s früher am Pfahlkopf detektiert. Der Unterschied ist allerdings so gering, dass beide Modelle als gleichwertig zu betrachten sind. Eine Betrachtung der maximalen Abweichung für die Beschleunigung, Geschwindigkeit und Verschiebung eines jeden Zeitschritts liefert $0,0076 \text{ ms}^{-2}$, $1,26 \cdot 10^{-6} \text{ ms}^{-1}$ und $2,9 \cdot 10^{-10} \text{ m}$. Die Maximal- und Minimalwerte der Beschleunigung, Geschwindigkeit und Verschiebung über den gesamten Berechnungszeitraum sind identisch. Die vorzeitige beziehungsweise verzögerte Reflexion der Welle am Pfahlfuß zieht somit lediglich eine leichte Phasenverschiebung nach sich. Bild 8.13 zeigt diese Abweichungen für die Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf der beiden Konfigurationen. Die dargestellten Δ -Werte werden mit $\sqrt{|\cdot|^2 - \cdot^2|}$ bestimmt. Damit sind die Abweichungen so klein, dass bei der Betrachtung der Ergebnisse im Weiteren keine Unterscheidung erfolgt.

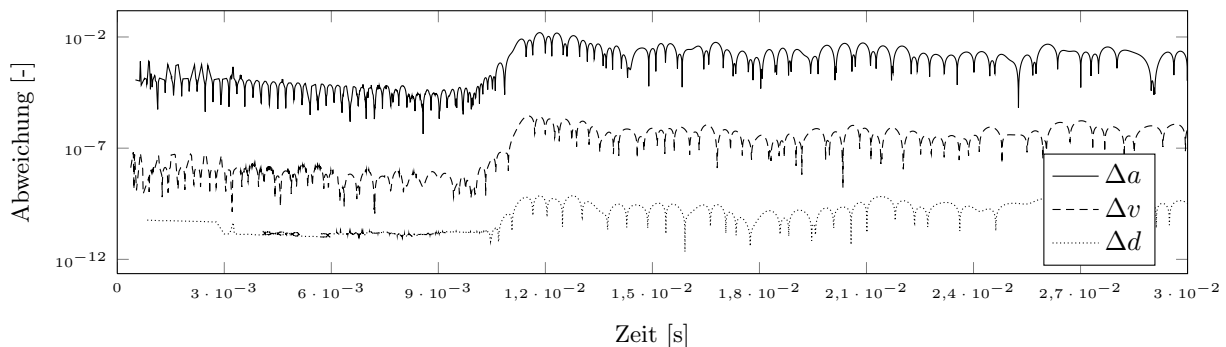


Bild 8.13.: Abweichung von Beschleunigung, Geschwindigkeit und Verschiebung am Pfahlkopf für die zwei Bodenkonfigurationen aus Bild 8.12.

Der erste Impedanzsprung beim Übergang vom freien Pfahlende zu dem im Boden eingebetteten Pfahl ist deutlich zu erkennen. Die durch den Impuls eingebrachte Welle benötigt etwa $2,65 \cdot 10^{-4}$ s vom Pfahlkopf bis zu diesem Punkt. Nach der doppelten Zeit wird die Teilreflexion am Pfahlkopf erfasst. Mit derselben Periode wiederholt sich der Vorgang, was die Oszillation im Funktionsverlauf erklärt. Nach $5,57 \cdot 10^{-3}$ s erreicht der Impuls den Pfahlfuß und wird auch hier zum Teil reflektiert. Da bei jeder Teilreflexion nur ein Teil der Wellen reflektiert wird, nimmt die Amplitude dieser Teilreflexion stetig ab, bis sie nicht mehr auszumachen ist. Begünstigt wird der Vorgang zusätzlich durch die bei der Berechnung eingebrachte numerische Dämpfung. Da der Boden unterhalb des Pfahlfußes dem Pfahl sehr ähnliche Materialparameter aufweist, geht ein Großteil der Energie in den Boden über. Bei genauer Betrachtung der am Pfahlkopf aufgezeichneten Informationen sind große Ähnlichkeiten mit der Fehlerkonfiguration drei und vier aus dem vorangegangenen Beispiel zu erkennen (vgl. Abschnitt 8.2.1.1 sowie 8.2.1.2). Auch hier kommt es zu einer Phasenumkehr, die in diesem Fall jedoch nicht auf eine Erweiterung des Pfahlquerschnittes zurückzuführen ist, sondern auf eine Kombination aus ähnlichen Materialparametern im Zusammenspiel mit einer sehr starken, wenn auch fiktiven, Querschnittserweiterung im Übergang vom Pfahl in den Boden.

8.3. Dynamische Lasten aufgrund von Verkehr

Das nachfolgende Beispiel ist sehr stark abstrahiert, zeigt jedoch weitere weitaus komplexere Einsatzmöglichkeiten des gekoppelten Verfahrens auf. Simuliert werden soll ein vorbeifahrender Zug, entlang dessen Gleisstrecke drei unterschiedliche Bauwerke platziert sind. Bild 8.14 zeigt den Aufbau des betrachteten Nahfeldes. Die Gebäude sind in Bild 8.14a von 1 bis 3 nummeriert. Bild 8.14b zeigt den Streckenverlauf von oben. Zusätzlich sind die Knotennummern des FE-Netzes dargestellt, auf die im weiteren Verlauf die zeitlich veränderlichen Lasten aufgebracht werden.

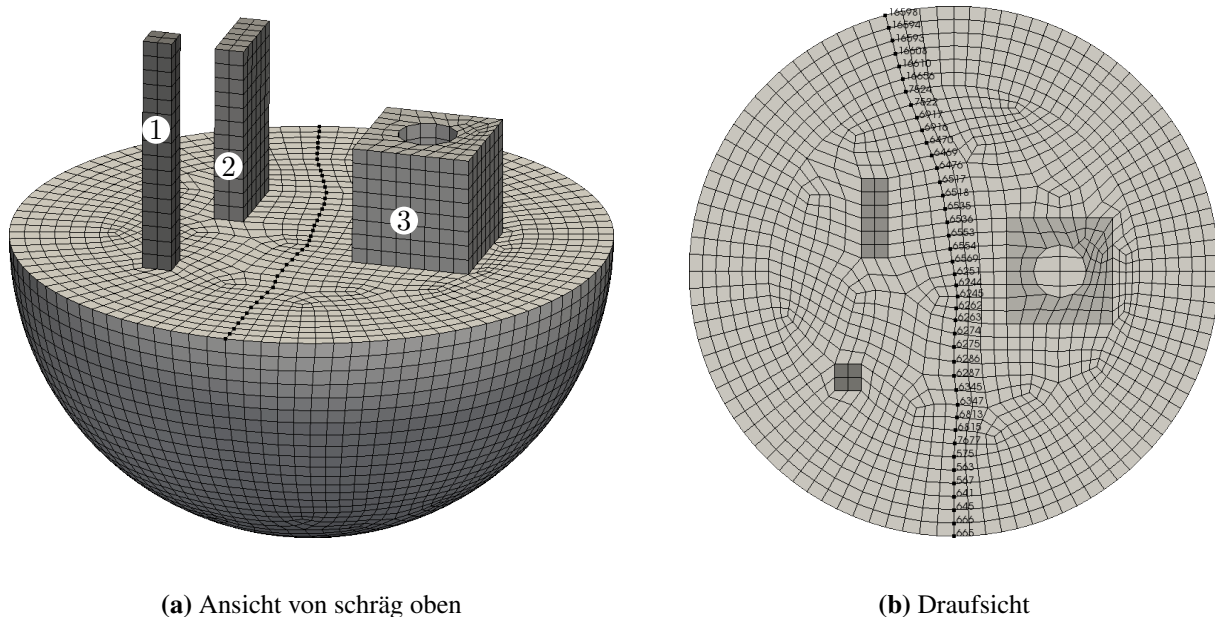


Bild 8.14.: Ansichten des Nahfeldes mit drei Bauwerken sowie eingezeichneter Gleisstrecke.

Die Länge des Zuges beträgt 30 m. Der Zug soll mit $33,33 \text{ ms}^{-1} = 120 \text{ kmh}^{-1}$ auf dem Gleis an den Gebäuden vorbeifahren. Aufgrund der gewählten Größe des Nahfeldes mit einem Radius von 100 m passiert der Zug das gesamte Gebiet innerhalb von sieben Sekunden. Für die Simulation werden zehn Sekunden angesetzt.

8.3.1. Modellbildung

Das Nahfeld, bestehend aus dem Boden und den drei in Bild 8.14 dargestellten Gebäuden, wird mit 18744 *Brick8* Elementen diskretisiert. Für Gebäude und Boden sind die Materialparameter in Tabelle 8.5 angegeben. Es sei an dieser Stelle darauf hingewiesen, dass die Gebäude als Volumenkörper gestaltet sind und somit sehr stark vereinfacht sind.

Gebäudefundamente und Gleis werden bei der Modellbildung nicht berücksichtigt. Die Modellierung des angekoppelten Fernfeldes erfolgt unter Verwendung der Substrukturtechnik, um den

	E-Modul MNm ⁻²	Poissonzahl	Dichte tm ⁻³
Gebäude 1	200	0,3	1,0
Gebäude 2	10	0,15	0,5
Gebäude 3	10	0,15	0,45
Boden	100	0,3	2,2

Tabelle 8.5.: Zusammenfassung der Materialkenngrößen.

Rechenaufwand zu reduzieren. Dafür wird das Fernfeld in drei Substrukturen aufgeteilt (vgl. Abschnitt 6.3). Darüber hinaus werden lediglich die ersten 200 Einflussmatrizen der Fernfelder bestimmt (vgl. Abschnitt 6.1). Alle weiteren Matrizen werden extrapoliert. Tabelle 8.6 fasst die Anzahl der Knoten, Elemente und Freiheitsgrade für das Nahfeld und für die Substrukturen des Fernfeldes zusammen.

Nahfeld FEM	Knoten	Elemente	Freiheitsgrade
Gebäude 1		64 <i>Brick8</i>	
Gebäude 2		144 <i>Brick8</i>	
Gebäude 3		512 <i>Brick8</i>	
Boden		18024 <i>Brick8</i>	
Σ	20989	18744	62967

Fernfeld SBFEM	Knoten	Elemente	Freiheitsgrade
Sub 1	833	768 <i>Quad4</i>	2499
Sub 2	767	716 <i>Quad4</i>	2301
Sub 3	901	832 <i>Quad4</i>	2703

Tabelle 8.6.: Anzahl der Elemente in Nahfeld und Fernfeld.

Als Grundlage für die Modellierung der Zuges wird eine Zuglänge von 30 m und ein Gesamtgewicht von 50 t angesetzt. Damit entspricht der Zug in etwa einem heutigen Regio-Shuttle, wie er im Nahverkehr eingesetzt wird. In der Simulation wird der Zug als verschiebbare Streckenlast berücksichtigt, die entlang der definierten Gleisstrecke (vgl. Bild 8.14b) örtlich verschoben wird. Um dies zu ermöglichen, ist die Lastfunktion

$$f_z^i = \frac{\ell_f p}{150\Delta t} \begin{pmatrix} t - t_0^i & \text{wenn } \{t_0^i \leq t \leq t_0^i + 150\Delta t\} \\ 150\Delta t & \text{wenn } \{t_0^i + 150\Delta t < t < t_0^i + 600\Delta t\} \\ t_0^i - t + 600\Delta t + 1 & \text{wenn } \{t_0^i + 600\Delta t \leq t < t_0^i + 750\Delta t\} \end{pmatrix} \quad (8.6)$$

für jeden Knoten i auszuwerten. Somit wird in jeden Knoten eine über die Zeit trapezförmige Last eingeleitet. Dabei wird über die Variable $\ell_f = 0,5$ für den ersten und letzten Punkt in

der betrachteten Strecke lediglich die halbe Last bestimmt. Für alle anderen Lastfunktionen gilt $\ell_f = 1$. Die Gewichtskraft des Zuges geht mit $p = \frac{1}{6} \cdot 50 \text{ t} \cdot 9,81 \text{ ms}^{-2}$ in die Lastfunktion ein. Die Masse des Zuges wird mit der Erdbeschleunigung multipliziert, um die Gewichtskraft zu bestimmen und auf die Anzahl der gleichzeitig belasteten Knoten aufzuteilen. Die Simulationszeit von 10 s wird mit einer Zeitschrittlänge von $\Delta t = 1 \cdot 10^{-3} \text{ s}$ realisiert. Aus der Zeitschrittlänge und der angenommenen Geschwindigkeit folgt die Anzahl an Zeitschritten, die der Zug von einem zum nächsten Knotenpunkt benötigt. Die $150\Delta t$ werden ebenfalls in der Lastfunktion berücksichtigt, um die Knoten von „unbelastet“ auf „vollbelastet“ und wieder zurück auf „unbelastet“ zu setzen. Ist die volle Belastung erreicht, wird sie für $450\Delta t$ aufgebracht, bevor die Last wieder reduziert wird. Der Zeitpunkt der Lasteinleitung ist durch den ortsabhängigen Parameter t_0^i gegeben, der die zeitliche Differenz zwischen dem Startpunkt und dem aktuellen Knoten beschreibt.

Zusätzlich zu dem aus Nahfeld und Fernfeld (FEM-SBFEM) bestehenden Modell wird das Nahfeld (FEM) ohne den angrenzenden Halbraum simuliert. In diesem Fall werden, wie schon bei dem Beispiel in Abschnitt 8.1 beschrieben, alle Randknoten mit den Koordinaten ($\sqrt{x^2 + y^2} = r, z = 0$) in allen drei Raumrichtungen festgehalten. Die Zeitschrittintegration wird in beiden Fällen mit dem HHT- α -Verfahren, nach Tabelle 3.1 durchgeführt. Neben der numerischen Dämpfung des Zeitschrittintegrationsverfahrens wird eine Rayleigh-Dämpfung mit $c_m = c_k = 5 \cdot 10^{-6}$ angesetzt.

8.3.2. Ergebnisse

Es sei im Voraus noch einmal darauf hingewiesen, dass das hier modellierte Beispiel sehr einfach gestaltet ist und für die Auslegung von Gebäudestrukturen, Fundamenten oder Ähnlichem nicht geeignet ist. Es dient hier lediglich der Demonstration bezüglich der Einsatzmöglichkeiten des gekoppelten Verfahrens.

Bild 8.15 zeigt die Verschiebungen der FEM-SBFEM Berechnung qualitativ. Dabei sind die berechneten Verschiebungen stark überhöht dargestellt. Es ist gut zu sehen, wie die Last durch das Berechnungsgebiet hindurch geschoben wird und dabei die umliegenden Bauwerke anregt. Nachdem die Last nach etwa 7 s das Nahfeld passiert hat, sind die Verschiebungen in den angeregten Gebäuden immer noch deutlich sichtbar.

Die Knotenpunktverschiebung je Raumrichtung sowie die Amplitude wird im Bild 8.16 für ausgewählte Knotenpunkte zusammengefasst. Dabei wird an jedem der drei Gebäude jeweils ein Knoten am Übergang vom Boden zum Bauwerk dargestellt. Die Knoten sind in Bild 8.16a markiert und die zeitlichen Verläufe der Verschiebungen in den Bildern 8.16b bis 8.16d über die Zeit aufgetragen. Im oberen Teil der Bilder sind jeweils die Ergebnisse der gekoppelten Simulation und im unteren Teil die Ergebnisse der FEM Simulation zu sehen. Die Anregung der Knoten erfolgt in Abhängigkeit ihrer Lage zur eingeleiteten Belastung. Zuerst wirkt sich die Last auf Knoten 424, dann auf Knoten 5789 und zuletzt auf den Knoten 115 aus. Damit entspricht die

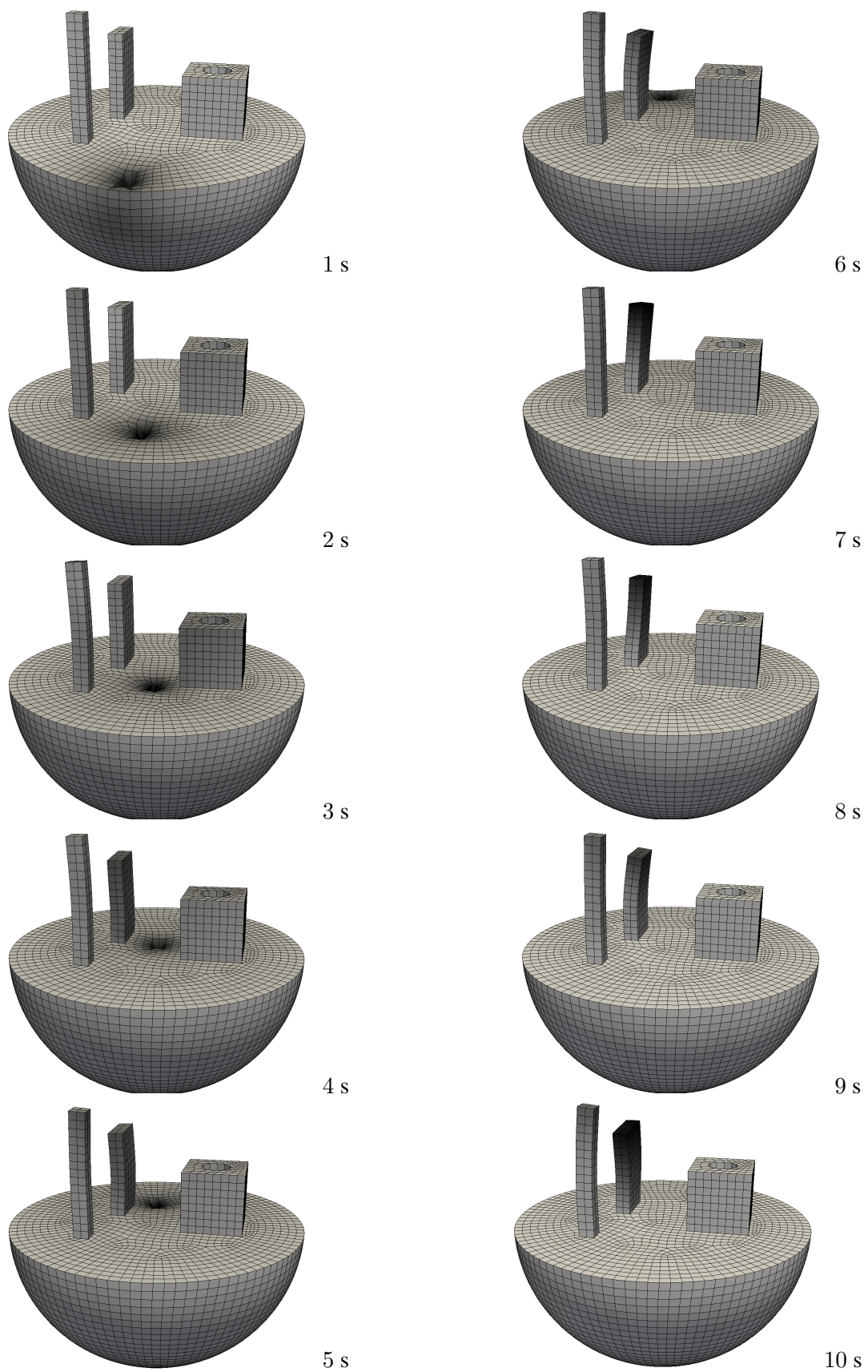
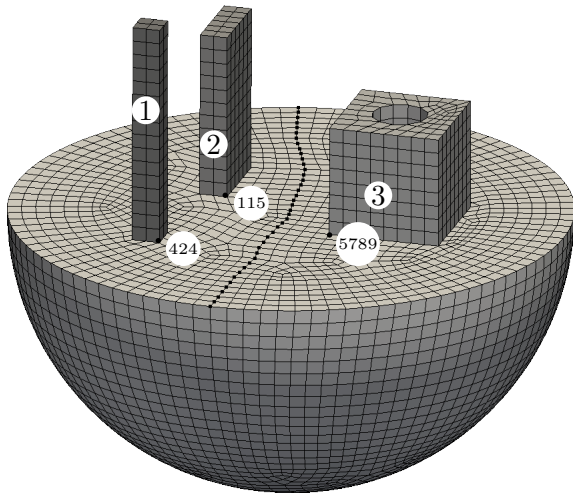
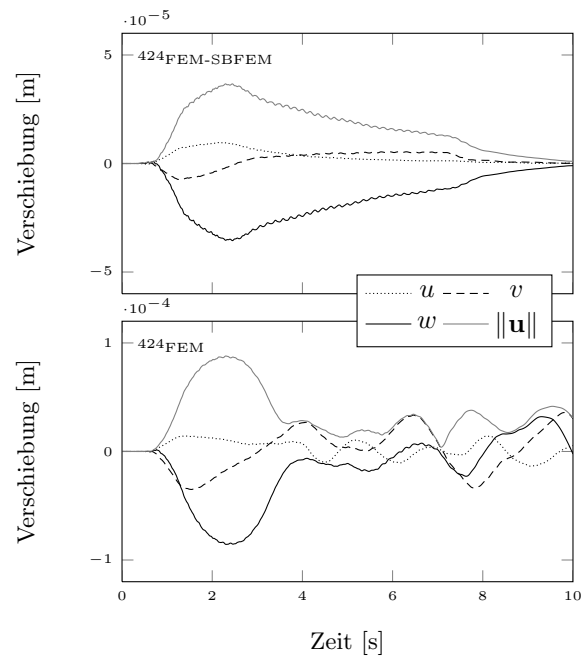


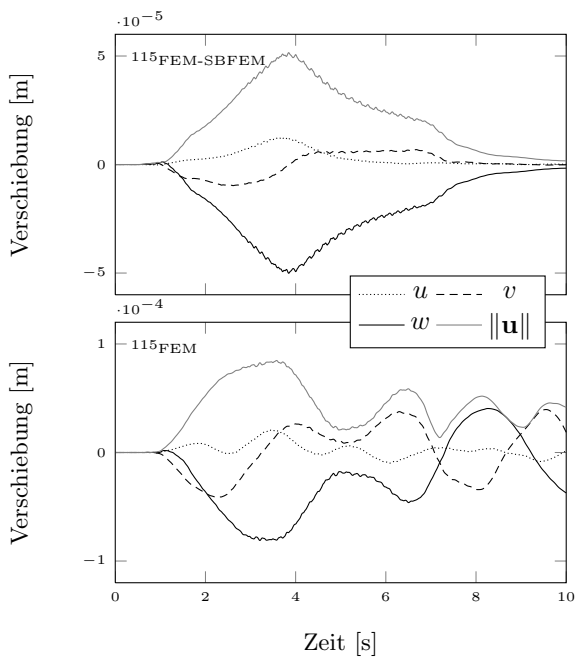
Bild 8.15.: Überhöhte Darstellung der zeitveränderlichen Verschiebungen der FEM-SBFEM Simulation.



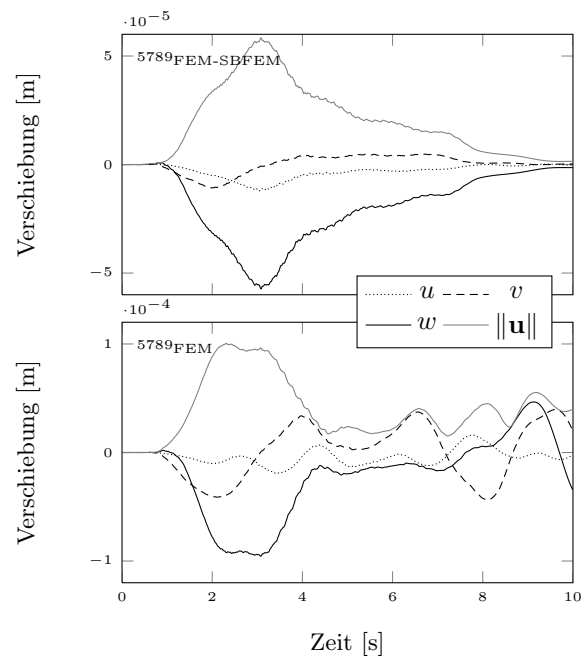
(a) Gebäude eins bis drei mit ausgewählten Knoten am Übergang vom Boden zum Gebäude.



(b) Verschiebungen am Knoten 424.



(c) Verschiebungen am Knoten 115.



(d) Verschiebungen am Knoten 5789.

Bild 8.16.: Zusammenstellung der räumlichen Knotenverschiebungen u , v , w und der zugehörigen Magnitude $\|u\|$ an ausgewählten Punkten der drei Gebäude für die Simulation des Nahfeldes (FEM) sowie die gekoppelte Berechnung aus Nah- und Fernfeld (FEM-SBFEM).

Man beachte die unterschiedliche Skalierung der Ordinate.

Simulation den Erwartungen. Im Falle der gekoppelten Simulation steigt die Verschiebungsamplitude mit der Vorbeifahrt des Zuges an, bis sie ihr Maximum erreicht hat. Nachdem der Zug den Punkt passiert hat, fällt diese danach erwartungsgemäß wieder auf den Ausgangswert. Je nach Abstand der Last zum Knoten fallen die sich einstellenden Verschiebungen unterschiedlich aus. In den drei Knoten stellt sich jedoch eine maximale Verschiebung von $3,67 \cdot 10^{-5}$ m bis $5,84 \cdot 10^{-5}$ m ein.

Bei der FEM-Berechnung stellen sich größere Verformungen als bei der gekoppelten FEM-SBFEM-Simulation ein. Hier liegen die Maxima der Verschiebungsamplituden bei einem Wert von $8,48 \cdot 10^{-5}$ m bis $1,01 \cdot 10^{-4}$ m und sind damit etwa doppelt so groß wie die der gekoppelten Simulation. Begründet ist dies durch die Überlagerung mit den am Diskretisierungsrand reflektierten Wellen, die die berechneten Verschiebungen verfälschen. Auch im späteren Verlauf der Simulation sind die Wellenüberlagerungen aufgrund der Reflexionen deutlich zu erkennen.

Derartige Simulationen könnten im Weiteren dazu genutzt werden, maximale Verschiebungen am Fundament oder bei detaillierterem Aufbau der Gebäude auch innerhalb der Gebäude zu bestimmen. Dies ist allerdings nur dann möglich, wenn die Boden-Bauwerk-Simulation plausible Ergebnisse liefert, was im Falle der hier gezeigten gekoppelten FEM-SBFEM-Simulation gegeben ist. Auch in diesem Beispiel ist der Vorteil der gekoppelten Simulation gegenüber einer reinen FEM-Simulation deutlich zu erkennen.

9. Zusammenfassung und Ausblick

Die numerische Simulation der Boden-Bauwerk-Interaktion erfordert eine korrekte Beschreibung der Wellenausbreitung sowohl im Nahfeld als auch im Fernfeld. Dabei muss das Fernfeld die ungestörte Wellenausbreitung in den unendlichen Halbraum sicherstellen können. Die Betrachtung von Bauwerk und Boden ist erforderlich, weil beide miteinander verbunden sind und somit miteinander interagieren.

Aktuell existieren eine Reihe von Verfahren, die zur Simulation der Boden-Bauwerk-Interaktion genutzt werden können. Hierzu zählen beispielsweise die Finite Elemente Methode, die Randelementemethode, die Integraltransformationemethode sowie die *Scaled Boundary Finite Element Method*. Um die jeweiligen Vorteile der einzelnen Methoden zu nutzen und Nachteile zu kompensieren, werden im Einzelfall unterschiedliche Verfahren miteinander gekoppelt.

In der vorliegenden Arbeit wird gezeigt, wie ein effizienter Ansatz aus gekoppelter Finite Elemente Methode und *Scaled Boundary Finite Element Method* entwickelt und genutzt werden kann, um geotechnische Fragestellungen unter Berücksichtigung des unendlichen Halbraums zu untersuchen. Bei dem gekoppelten Ansatz wird das Nahfeld, das die Struktur sowie den anstehenden Baugrund beinhaltet, mit der Finite Elemente Methode und das Fernfeld, das den umgebenden unendlichen Halbraum beschreibt, mit der *Scaled Boundary Finite Element Method* diskretisiert. Mit dem vorgestellten numerischen Modell ist es möglich, realitätsnahe Simulationen der Boden-Bauwerk-Interaktion unter Berücksichtigung einer großen Anzahl an Freiheitsgraden in Nahfeld und Fernfeld in angemessener Zeit zu realisieren.

Aufgrund der gewählten Formulierung ist zur Berechnung des Fernfeldeinflusses ein Faltungsintegral in jedem Zeitschritt zu lösen, was mit einem hohen Berechnungsaufwand einhergeht. Um den hohen Berechnungsaufwand und gleichzeitig den erforderlichen Speicherbedarf zu reduzieren, werden drei unterschiedliche Möglichkeiten zur Modellreduktion des Fernfeldes gezeigt und ausführlich diskutiert. Aufgrund der Tatsache, dass die Einträge in den Einflussmatrizen ab einem gewissen Zeitpunkt um einem konstanten Wert anwachsen, ist es möglich die Matrizen nachfolgender Zeitschritte auf Grundlage bekannter, bereits berechneter Matrizen zu extrapolieren. Erfolgt die Extrapolation der Einflussmatrizen wie hier unter Verwendung eines rekursiven Algorithmus, dann sind Reduktionen von Rechenzeit und Speicherbedarf auf etwa 30% leicht möglich, ohne die Genauigkeit der Simulation zu beeinflussen. Eine weitere Modellreduktion zielt auf die geometrische Entkopplung der Randknoten. In dem Fall werden nach vollständiger Berechnung der Einflussmatrizen nur die Werte für die Kopplung von Nahfeld und Fernfeld berücksichtigt, die oberhalb einer definierten Schranke liegen. Durch dieses Vorgehen kann der benötigte Speicher massiv reduziert werden. Der Einfluss auf die Rechenzeit ist, wie gezeigt wird, gering. Als dritte Möglichkeit der Modellreduktion wird die Zerlegung des Fernfeldes in Substrukturen untersucht. Die Zerlegung erfordert zusätzlichen Aufwand bei der Netzerzeugung. Speicherbedarf und Rechenzeit können so jedoch signifikant reduziert werden. Bei Kombination aller drei Modellreduktionen kann, bei den hier diskutierten Beispielen, ge-

zeigt werden, dass die benötigte Rechenzeit auf 6% reduziert werden kann. Der Speicherbedarf liegt in dem Fall bei gerade einmal 11% der Ausgangskonfiguration.

Da die Simulation der Boden-Bauwerk-Interaktion in der Regel mit einem sehr hohen Rechenaufwand einhergeht, werden die Programme derart umgesetzt, dass sie auf parallelen Systemen mit verteiltem Speicher einsetzbar sind. Bei der Implementierung der Programme SCABO und ELPASO wird auf externe Bibliotheken zurückgegriffen. Dies ermöglicht effiziente Programme zu entwickeln, erfordert aber zugleich eine sehr sorgfältige Umsetzung bei der Programmierung.

Die Leistungsfähigkeit der implementierten Programme wird diskutiert. Wobei das Programm SCABO, das zur Bestimmung der Einflussmatrizen des Fernfeldes eingesetzt wird, und das Programm ELPASO, mit dem der gekoppelte FEM-SBFEM-Ansatz gelöst wird, eingehend untersucht werden. Die Effizienz der auf verteilten Systemen arbeitenden Programme wird gezeigt. Auf der Grundlage der starken Skalierung kann ein sequentieller Anteil in den Programmen SCABO und ELPASO von 2,5% bis 5% abgeschätzt werden. Was einem *Speedup* von 20 bis 40 entspricht. Ein Problem konstanter Größe wird auf unterschiedlich vielen Prozessen gelöst und die benötigte Rechenzeit mit der Referenzlösung aus der sequentiellen Berechnung verglichen. Werden die diskretisierten Gebiete mit Anzahl der Prozesse größer, zeigt sich bei der schwachen Skalierung ein noch besseres Skalierungsverhalten. Für diesen Fall kann der sequentielle Programmanteil bei etwa 0,05% bis 1% abgeschätzt werden, das entspricht einem theoretisch möglichen *Speedup* von 100 bis 200.

Für eine effiziente Simulation des gekoppelten FEM-SBFEM Ansatzes sind somit zwei wesentliche Punkte zu beachten. Zum einen ist das Fernfeld hinsichtlich der vorgestellten Modellreduktionsverfahren zu vereinfachen, um im Vorfeld den Berechnungsaufwand und Speicherbedarf zu reduzieren. Zum anderen ist trotz der Modellreduktion des Fernfeldes wegen des immer noch hohen Rechenaufwands und Speicherbedarfs eine Parallelisierung der Berechnungssoftware nötig, um realitätsnahe Modelle untersuchen zu können. Diese Parallelisierung ist aus heutiger Sicht bedingt durch die Architektur aktueller Computer zeitgemäß.

Einige Anwendungsbeispiele, für die die entwickelten Programme heute bereits eingesetzt werden können, werden gezeigt. Damit legt diese Arbeit die nötigen Grundlagen für zukünftige Untersuchung ähnlicher oder auch weitaus komplexerer Problemstellungen. Diese Arbeit stellt dabei einen wesentlichen Beitrag in diese Richtung dar.

Zur weiteren Steigerung der Effizienz der vorgestellten Programme können zwei unterschiedliche Wege verfolgt werden. Zum einen ist es möglich zusätzliche Modellreduktionen zu integrieren und bei der Berechnung zu berücksichtigen. So könnten beispielsweise zwei unterschiedliche Zeitskalen im Nahfeld und Fernfeld berücksichtigt werden. Dies würde die Anzahl der benötigten Einflussmatrizen des Fernfeldes weiter reduzieren können. Fernfeldinformationen sind dann nicht nur, wie in der Arbeit vorgestellt, zu extrapolieren, sondern auch zwischen den Zeitschritten zu interpolieren. Welchen Einfluss die Verwendung zweier unterschiedlicher

Zeitschrittlängen aus die Lösung der Simulation hat und ob andere Modellreduktionen kombiniert werden können, ist zu untersuchen. Zum anderen ist eine hybride Implementierung der Programme SCABO und ELPASO möglich, bei der neben der Bibliothek MPI auch OpenMP genutzt wird, wodurch eine Anpassung an die aktuell vorliegende Computerarchitektur, bei der eine CPU mehrere *Core* enthält, erfolgen würde. Prozesse, die auf verteilte Speicher zugreifen müssen, nutzen, wie bereits implementiert, MPI für die Kommunikation. Prozesse, die sich einen gemeinsamen Speicher teilen, können OpenMP nutzen und müssen nicht mehr über MPI kommunizieren. Eine hybride Implementation verspricht aufgrund der Aufteilung der Datenströme sowie verkürzter Zugriffszeiten innerhalb des gemeinsam genutzten Arbeitsspeichers eine weitere Steigerung der Effizienz. Gegebenenfalls können weitere Bibliotheken hinzugefügt werden, die spezielle Aufgaben übernehmen und somit ebenfalls die Effizienz der Programme steigern können.

Aktuell erfolgt die Berechnung der Einflussmatrizen vorab. Erst nach vollständiger Berechnung der Einflussmatrizen wird die gekoppelte Simulation aus FEM und SBFEM durchgeführt. Alternativ könnte die Berechnung der Einflussmatrizen auch zeitgleich, während der gekoppelten Simulation, erfolgen. Somit wäre es möglich auf die zeitintensiven Schreib- und Leseprozesse auf die Festplatte verzichten zu können, da bereits alle Daten im Arbeitsspeicher vorhanden wären. Das Schreiben und Lesen der Einflussmatrizen wäre dann nur noch erforderlich, wenn mehrere Nahfeldkonfigurationen mit demselben Fernfeld gekoppelt werden sollen.

Je nach Anwendungsbereich müssen die Algorithmen um höherwertige Materialgesetze zur Darstellung großer Verformungen oder Berücksichtigung endlicher Spannungen ergänzt werden. Die Erweiterungen sind für das Nahfeld und für die Verwendung der FEM bereits gelöst. Sollte es zu einem nicht-linearen Verhalten des Nahfeldes im Bereich des Kopplungsrandes kommen, ist das Fernfeld und damit die SBFEM ebenfalls um höherwertige Ansätze zu ergänzen. Dies ist für die Ausbreitung dreidimensionaler Wellen im Zeitbereich für die SBFEM bisher nicht gelöst.

A. Nomenklatur

A	Fläche	ℓ	Länge
B_1, B_2	Operatormatrix	ℓ_{el}	Elementlänge
C_1, C_2, C_3	Koeffizientenmatrix	M	Massenmatrix
C	Dämpfungsmatrix	M^∞	Einflussmatrix
c_k	steifigkeitsproportionaler Dämpfungsparameter	N	Ansatzfunktionsmatrix
c_m	masseproportionaler Dämpfungsparameter	N	Ansatzfunktion
c_p	Kompressionswellen- geschwindigkeit	p	Lastvektor
c_s	Transversalwellen- geschwindigkeit	\hat{p}	effektiver Lastvektor
c_R	Rayleigh-Wellengeschwindigkeit	p_∞	Disipationsparameter
c_L	Love-Wellengeschwindigkeit	p	Prozesse
D	Operatormatrix	q	Flächenlast
E	Elastizitätsmatrix	r	Radius
E	Elastizitätsmodul	S^∞	dynamische Steifigkeitsmatrix
E_p	parallele Effizienz	S_p	paralleler <i>Speedup</i>
E'_p	skalierte Effizienz	S'_p	skalierter <i>Speedup</i>
E_{ext}	externe Energie	$S_{p,max}$	maximaler <i>Speedup</i>
E_{kin}	kinetische Energie	s_{ij}	Deviatorischer Spannungstensor
E_{pot}	potentielle Energie	s	Setzung
E_{tot}	Gesamtenergie	T^∞	Gradienten der Einflussmatrix $\frac{\Delta M^\infty}{\Delta t}$
F	Kraft	T	Periode, Zeitpunkt, Laufzeit
f	Kraftvektor	t	Spannungsvektor
f	Frequenz	t_n	Zeit
f_{max}	maximale Frequenz	t_m	aktueller Zeitschritt
G	Schubmodul	t_{krit}	Zeitpunkt der Extrapolation
I	Einheitsmatrix	u	kritische Zeitschrittlänge
I_i	Spannungsinvarianten	u_0	Verschiebungsvektor
J	Jacobimatrix	\dot{u}	Anfangswertbedingung für den Verschiebungsvektor
J_i^D	Spannungsinvarianten des Deviators	\dot{u}_0	Geschwindigkeitsvektor
K	Steifigkeitsmatrix	\ddot{u}	Anfangswertbedingung für den Geschwindigkeitsvektor
\hat{K}	effektive Steifigkeitsmatrix	u, v, w	Beschleunigungsvektor
K	Kompressionsmodul	v	Verschiebungen
k	Wellenzahl	x	Geschwindigkeit
L	untere Cholesky Dreiecksmatrix	x, y, z	globaler Koordinatenvektor
			Raumkoordinaten

Z	Impedanz	λ	erste Lamé-Konstante
		μ	zweite Lamé-Konstante
α	sequentieller Programmanteil	ν	Querdehnzahl
$\alpha, \alpha_f, \alpha_m$	Zeitschrittintegrationsparameter	ξ	lokaler Koordinatenvektor
β	Zeitschrittintegrationsparameter	ξ_1, ξ_2, ξ_3	Elementkoordinaten FEM
Γ	Rand des Berechnungsgebiets	ξ, η, ζ	Elementkoordinaten SBFEM
γ	Zeitschrittintegrationsparameter	ρ	Dichte
Δt	Zeitschrittlänge	σ	Spannungstensor
Δ	Laplace-Operator $\sum_{k=1}^n \frac{\partial^2}{\partial x_k^2}$	σ_{ij}	Spannungskomponenten
δu	virtuelle Verschiebung	σ_m	hydrostatischer Spannungstensor
δW	virtuelle Arbeit	σ_i	Hauptspannungen
$\delta \epsilon$	virtuelle Dehnung	σ_m	mittlere Normalspannung
ϵ	Dehnungstensor	σ_v	Vergleichsspannung
ϵ_{ij}	Dehnungskomponenten	ς	Dämpfungsgrad
ϵ_z	geometrische Toleranz	τ_{ij}	Subspannungskomponenten
κ	Dämpfungsparameter	ψ	Verhältniswert
λ	Wellenlänge	Ω	Berechnungsgebiet
λ_{\max}	maximale Wellenlänge	ω	Eigenfrequenz

BEM	<i>Boundary Element Method</i> , Randelementmethode
BLACS	<i>Basic Linear Algebra Communication Subprograms</i>
BLAS	<i>Basic Linear Algebra Subprograms</i>
CPU	<i>Central Processing Unit</i>
ELPASO	<i>elementary Parallel Solver</i>
EVA	Eingabe-Verarbeitung-Ausgabe
FEM	Finite Elemente Methode
HHT- α	HILBER-HUGHES-TAYLOR- α
LAPACK	<i>Linear Algebra PACKage</i>
MIMD	<i>Multiple Instruction Multiple Data</i>
MPI	<i>Message Passing Interface</i>
MPMD	<i>Multiple Program Multiple Data</i>
PBLAS	<i>Parallel Basic Linear Algebra Subprograms</i>
PETSc	<i>Portable, Extensible Toolkit for Scientific Computation</i>
PVM	<i>Parallel Virtual Maschine</i>
SBFEM	<i>Scaled Boundary Finite Element Methode</i>
SCABO	<i>Scaled Boundary</i>
ScaLAPACK	<i>Scalable Linear Algebra PACKage</i>
SLICOT	<i>Subroutine Library in Systems and Control Theory</i>
WBZ- α	WOOD-BOSSAK-ZIENKIEWICZ- α

Literaturverzeichnis

- [1] Amdahl, Gene M.: [Validity of the single processor approach to achieving large-scale computing capabilities](#). *AFIPS Conference Proceedings*, 30, 483–485, 1967.
- [2] Amdahl, Gene M.; Rattner, Justin: [Validity of the single processor approach to achieving large-scale computing capabilities – Reprinted from the AFIPS Conference Proceedings](#). *IEEE SSCS NEWS*, 12(3), 19–20, 2007.
- [3] Anderson, E.; Bai, Z.; Bischof, C.; Demmel, J.; Dongarra, J.; Croz, J. Du; Greenbaum, A.; Hammarling, S.; McKenney, A.; Sorensen, D.: *LAPACK User's Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1992.
- [4] Antes, H.; Spyrakos, C.: Soil-Structure Interaction In *Computer Analysis and Design of Earthquake Resistant Structures*. (Beskos, D.E.; Anagnostopoulos, S.A., Eds.), Computational Mechanics Publications, Southampton, 271–332, 1997.
- [5] Appelö, Daniel; Colonius, Tim: [A high-order super-grid-scale absorbing layer and its application to linear hyperbolic systems](#). *Journal of Computational Physics*, 228(11), 4200–4217, 2009.
- [6] Astley, R. J.: [Infinite elements for wave problems: a review of current formulations and a assessment of accuracy](#). *International Journal for Numerical Methods in Engineering*, 49(7), 951–976, 2000.
- [7] Balay, Satish; Brown, Jed; ; Buschelman, Kris; Eijkhout, Victor; Gropp, William D.; Kaushik, Dinesh; Knepley, Matthew G.; McInnes, Lois Curfman; Smith, Barry F.; Zhang, Hong: PETSc Users Manual. Technical report ANL-95/11 - Revision 3.2, Argonne National Laboratory, 2011.
- [8] Balay, Satish; Brown, Jed; Buschelman, Kris; Gropp, William D.; Kaushik, Dinesh; Knepley, Matthew G.; McInnes, Lois Curfman; Smith, Barry F.; Zhang, Hong: PETSc Web page. <http://www.mcs.anl.gov/petsc>, 2011.
- [9] Balay, Satish; Gropp, William D.; McInnes, Lois Curfman; Smith, Barry F.: Efficient Management of Parallelism in Object Oriented Numerical Software Libraries In *Modern Software Tools in Scientific Computing*. (Arge, E.; Bruaset, A. M.; Langtangen, H. P., Eds.). Birkhäuser Press, 163–202, 1997.
- [10] Bargel, Hans-Jürgen; Schulze, Günter: *Werkstoffkunde*. Springer Vieweg, Berlin, 2012.
- [11] Bartsch, Hans-Jochen: *Taschenbuch mathematischer Formeln*, 20., *neu bearbeitete und erweiterte Auflage*. Fachbuchverlag Leipzig im Carl Hanser Verlag, München, Wien, 2004.

- [12] Basu, Ushnish; Chopra, Anil K.: [Perfectly matched layers for time-harmonic elastodynamics of unbounded domains: theory and finite-element implementation](#). *Computer Methods in Applied Mechanics and Engineering*, 192, 1337–1375, 2003.
- [13] Basu, Ushnish; Chopra, Anil K.: [Perfectly matched layers for transient elastodynamics of unbounded domains](#). *International Journal for Numerical Methods in Engineering*, 59, 1039–1074, 2004.
- [14] Bathe, Klaus-Jürgen: *Finite-Elemente-Methode*, Vol. 2., vollständig neu bearbeitete und erweiterte Auflage. Springer, Berlin, Heidelberg, 2002.
- [15] Bazyar, Mohammad Hossein; Song, Chongmin: [Time-harmonic response of non-homogeneous elastic unbounded domains using the scaled boundary finite-element method](#). *Earthquake Engineering and Structural Dynamics*, 35, 357–383, 2006.
- [16] Beck, S. C.; Langer, S.: Modellierung von strömungsinduzierter Schalleinleitung in poroelastische Materialien. *Proceedings of DAGA 2012 Deutsche Gesellschaft für Akustik e. V. (DEGA)*, 927–928, 2012.
- [17] Beck, Silja; Clasen, Dirk; Lehmann, Lutz; Rurkowska, Katherina; Schauer, Marco; Wulkau, Meike: ELPASO Manual. Technical report Revision: 433, TU Braunschweig, Institut für Angewandte Mechanik (2008–2013), Institut für Konstruktionstechnik (seit 2013), 2008–2014. <https://www.tu-braunschweig.de/ik>.
- [18] Beck, Silja C.: *Strömungsinduzierter Körperschalleintrag in Strukturen mit porösen Oberflächen*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 2012.
- [19] Beck, Silja C.; Langer, Sabine: Flow-induced Sound Radiation from Air-ducting Structures. *Proceedings of the International Conference on Acoustics NAG/DAGA Nederlands Akoestisch Genootschap (NAG) und Deutsche Gesellschaft für Akustik e. V. (DEGA)*, 1496–1498, 2009.
- [20] Beck, Silja C.; Wulkau, Meike; Langer, Sabine: Hybrid Numerical Simulation of Flow-Induced Radiation of Air-Ducting Structures Lined with Porous Material. *Proceedings of Pan-American Congress of Applied Mechanics - PACAM*, XI, 2009.
- [21] Beer, G.: *Programming the Boundary Element Method*. Wiley & Sons, Chichester, UK, 2001.
- [22] Benner, P.: Numerical Solution of Special Algebraic Riccati Equations via an Exact Line Search Method. In *Proc. European Control Conf. ECC 97, Paper 786*, Waterloo (B), BELWARE Information Technology, 1997.
- [23] Benner, P.; Byers, R.; Quintana-Ortí, E.S.; Quintana-Ortí, G.: [Solving Algebraic Riccati Equations on Parallel Computers Using Newton’s Method with Exact Line Search](#). *Parallel Comput.*, 26(10), 1345–1368, 2000.

- [24] Benner, P.; Mehrmann, V.; Sima, V.; Van Huffel, S.; Varga, A.: SLICOT - A Subroutine Library in Systems and Control Theory In *Applied and Computational Control, Signals and Circuits*. (Datta, B. N., Ed.), Vol. 1, Birkhäuser, 499–539, 1999.
- [25] Benner, P.; Quintana-Ortí, E. S.; Quintana-Ortí, G.: [Solving Linear-Quadratic Optimal Control Problems on Parallel Computers](#). *Optimization Methods and Software*, 23(6), 879–909, 2008.
- [26] Beskos, Dimitri E.: *Computational Methods in Mechanics: Volume 3: Boundary Element Methods in Mechanics*. Elsevier Science Publishers B.V., Amsterdam, 1987.
- [27] Bettess, P.: *Infinite Elements*. Penshaw Press, Sunderland, U.K., 1992.
- [28] Bettess, P.; Zienkiewicz, O. C.: [Diffraction and refraction of surface waves using finite and infinite elements](#). *International Journal for Numerical Methods in Engineering*, 11(8), 1271–1290, 1977.
- [29] Blackford, L. S.; Choi, J.; Cleary, A.; D’Azevedo, E.; Demmel, J.; Dhillon, I.; Dongarra, J.; Hammarling, S.; Henry, G.; Petitet, A.; Stanley, K.; Walker, D.; Whaley, R. C.: *ScaLAPACK Users’ Guide*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 1997.
- [30] BLACS: Basic Linear Algebra Communication Subprograms. <http://www.netlib.org/blacs>.
- [31] BLAS: Basic Linear Algebra Subprograms. <http://www.netlib.org/blas>.
- [32] Borsutzky, Robert: *Seismic Risk Analysis of Buried Lifelines*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 2008.
- [33] Borsutzky, Robert; Lehmann, Lutz: . Numerische Simulation unterirdischer Versorgungsleitungen unter Erdbebeneinwirkung. Internationales Graduiertenkolleg 802 „Risikomanagement bei Natur- und Zivilisationsgefahren für Bauwerke und Infrastrukturanlagen“, TU Braunschweig, 2005.
- [34] Bransch, Martin: *Elasto-plastic analysis with a coupled FEM/SBFEM approach*. Dissertation, Technische Universität Braunschweig, 2012.
- [35] Bransch, Martin; Lehmann, Lutz: [Simulation der Boden-Bauwerk-Interaktion unter Berücksichtigung eines elastisch-plastischen Materialverhaltens des Baugrundes](#). *Bautechnik*, 88(4), 237–243, 2011.
- [36] Bransch, Martin; Lehmann, Lutz: [A nonlinear HHT- \$\alpha\$ method with elastic–plastic soil–structure interaction in a coupled SBFEM/FEM approach](#). *Computers and Geotechnics*, 38(1), 80–87, 2012.
- [37] Brebbia, C. A.; Telles, J. C. F.; Wrobel, L. C.: *Boundary Element Techniques - Theory and Applications in Engineering*. Springer, Berlin, Heidelberg, 1984.

- [38] Brikert, Arndt; Haage, Stefan; Straub, Markus: *Umformtechnische Herstellung komplexer Karosserieteile: Auslegung von Ziehanlagen*. Springer Vieweg, Berlin, Heidelberg, 2013.
- [39] Bronstein, I. N.; Semendjajew, K. A.; Musiol, G.; Mühlig, H.: *Taschenbuch der Mathematik*. Verlag Harri Deutsch, Thun und Frankfurt am Main, 2001.
- [40] Chen, D.; Birk, C.; Song, C.; Du, C.: [A high-order approach for modelling transient wave propagation problems using the scaled boundary finite element method](#). *International Journal for Numerical Methods in Engineering*, 97(13), 937–959, 2013.
- [41] Chidgzy, Steven R.; Deeks, Andrew J.: [Determination of coefficients of crack tip asymptotic fields using the scaled boundary finite element method](#). *Engineering Fracture Mechanics*, 72(13), 2019–2036, 2005.
- [42] Chung, J.; Hulbert, G. M.: [A Time Integration Algorithm for Structural Dynamics with Improved Numerical Dissipation: The Generalized- \$\alpha\$ Method](#). *Journal of Applied Mechanics*, 60, 371–375, 1993.
- [43] Clasen, Dirk: *Numerische Untersuchung der akustischen Eigenschaften von trennenden und flankierenden Bauteilen*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 2008.
- [44] Crouch, Roger S.; Bennett, Terry: Efficient EBE Treatment of the Dynamic Far-Field in Non-Linear FE Soil-Structure Interaction Analyses. *European Congress on Computational Methods on Applied Sciences and Engineering ECCOMAS 2000*, 2000.
- [45] Cuthill, E.; McKee, J.: [Reducing the Bandwidth of Sparse Symmetric Matrices](#). In *Proceedings of the 1969 24th National Conference*, ACM '69, New York, NY, USA, ACM, 157–172, 1969.
- [46] de Araújo, Francisco Célio: *Zeitbereichslösung linearer dreidimensionaler Probleme der Elastodynamik mit einer gekoppelten BE/FE-Methode*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 1994.
- [47] Deutsche Gesellschaft für Akustik e. V.: *Akustische Wellen und Felder*, DEGA-Empfehlung 101. Berlin, 2006.
- [48] Dominguez, J.: *Boundary Elements in Dynamics*. Computational Mechanics Publications, Southampton, Boston, & Elsevier Applied Science, London, New York, 1993.
- [49] Dongarra, J. J.; Whaley, R. C.: LAPACK Working Note 94: A User's Guide to the BLACS v1.1. Technical report UT-CS-95-281, Department of Computer Science, University of Tennessee, 1995.
- [50] Dongarra, Jack J.; Duff, Iain S.; Sorensen, Danny C.; van der Vorst, Henk A.: *Numerical Linear Algebra for High-Performance Computers*. SIAM, Society for Industrial and Applied Mathematics, Philadelphia, 1993.

- [51] dpa/bar: Dom könnte wegen Lage der Schatzkammer zittern. <http://www.welt.de/regionales/koeln/article112700167/Dom-koennte-wegen-Lage-der-Schatzkammer-zittern.html>. welt-online.
- [52] EA-Pfähle: *Empfehlungen des Arbeitskreises Pfähle*. Deutsche Gesellschaft für Geotechnik, Ernst & Sohn, Dortmund, 2007.
- [53] Engquist, B.; Majda, A.: Absorbing boundary conditions for the numerical simulation of waves. *Mathematics of Computation*, 31(139), 629–651, 1977.
- [54] F.A.Z.: U-Bahn lässt den Kölner Dom erzittern. <http://www.faz.net/aktuell/gesellschaft/koeln-u-bahn-laesst-den-koelner-dom-erzittern-12019538.html>. Frankfurter Allgemeine Zeitung.
- [55] FFTW: Fastest Fourier Transform in the West. <http://www.fftw.org>.
- [56] Filonenko-Boroditsch, M. M.: *Elastizitätstheorie*. Übersetzung aus dem Russischen: E. Sander und H. Geißler, VEB Fachbuchverlag, Leipzig, 1967.
- [57] Fischer, J.; Missal, C.; Breustedt, M.; Stahlmann, J.: [Numerical simulation of low strain integrity tests on model piles](#). *NUMGE2010, 7th European Conference on Numerical Methods in Geotechnical Engineering*, 655–660, 2010.
- [58] Freedman, J. M.; Keer, L. M.: [Vibratory Motion of a Body on an Orthotropic Half Plane](#). *Journal of Applied Mechanics*, 39(4), 1033–1040, 1972.
- [59] Frühe, Georg: *Überlagerung von Grundlösungen in der Elastodynamik zur Behandlung der dynamischen Tunnel-Boden-Bauwerk-Interaktion*. Dissertation, Technische Universität München, Shaker-Verlag, Aachen, 2011.
- [60] Garelli, Luciano; Schauer, Marco: Two and three dimensional coupling concept of PETSc-FEM and eIPaSo for fluid structure interaction analysis. Technical report, TU Braunschweig, Institut für Angewandte Mechanik, 2013.
- [61] Garelli, Luciano; Schauer, Marco; D’Elia, Jorge; Storti, Mario A.; Langer, Sabine C.: Coupling Concept of two Parallel Research Codes for Two and Three Dimensional Fluid Structure Interaction Analysis. *International Journal of Mathematical, Computational, Physical and Quantum Engineering*, 8(5), 763–769, 2014.
- [62] Gau, L.; Kögl, M.; Wagner, M.: *Boundary Element Methods for Engineers and Scientists - An Introductory Course with Advanced Topics*. Springer, Berlin, Heidelberg, 2003.
- [63] Geist, A.; Beguelin, A.; Dongarra, J.; Jiang, W.; Manchek, R.; Sunderam, V.: *PVM: Parallel Virtual Machine - A Users’ Guide and Tutorial for Networked Parallel Computing*. MIT Press, Massachusetts Institute of Technology, 1994.

- [64] Genes, M.C.: [Dynamic analysis of large-scale SSI systems for layered unbounded media via a parallelized coupled finite-element/boundary-element/scaled boundary finite-element model](#). *Engineering Analysis with Boundary Elements*, 36, 845–857, 2012.
- [65] Genes, M.C.; Kocak, S.: A combined finite element based soil-structure interaction model for large-scale systems and applications on parallel platforms. *Engineering Structures*, 24, 1119–1131, 2002.
- [66] Givoli, Dan: [Non-reflecting Boundary Conditions](#). *Journal of Computational Physics*, 94, 1–29, 1991.
- [67] Givoli, Dan: *Numerical Methods for Problems in Infinite Domains*. Elsevier Science Limited, Amsterdam, 1992.
- [68] Granat, Robert; Kågström, Bo: Algorithm 904: The SCASY Library – Parallel Solvers for Sylvester-Type Matrix Equations with Applications in Condition Estimation, Part II. *ACM Transactions on Mathematical Software*, 37(3), 33:1–33:4, 2010.
- [69] Granat, Robert; Kågström, Bo: Parallel Solvers for Sylvester-Type Matrix Equations with Applications in Condition Estimation, Part I: Theory and Algorithms. *ACM Transactions on Mathematical Software*, 37(3), 32:1–32:32, 2010.
- [70] Granat, Robert; Kågström, Bo: SCASY - Parallel ScaLAPACK-style High Performance Library for Sylvester-Type Matrix Equations. <http://www8.cs.umu.se/granat/scasy.html>.
- [71] Gross, Dietmar; Seelig, Thomas: *Umformtechnische Herstellung komplexer Karosserieteile: Auslegung von Ziehanlagen*. Springer, Heidelberg, Dordrecht, London, New York, 2011.
- [72] Gross, Ditmar; Hauger, Werner; Wriggers, Peter: *Technische Mechanik 4 – Hydromechanik, Elemente der Höheren Mechanik, Numerische Methoden*. Springer-Verlag, Berlin, Heidelberg, 2012.
- [73] Grote, M. J.; Keller, J. B.: Exact Nonreflecting Boundary Condition For Elastic Waves. Technical report 98-08, Seminar für Angewandte Mathematik, Eidgenössische Technische Hochschule Zürich, 1998.
- [74] Gustafson, John L.: [Reevaluating Amdahl's law](#). *Communications of the ACM*, 31(5), 532–533, 1988.
- [75] Hackbuch, W.; Schwarz, H.R.; Zeidler, E.: *Teubner – Taschenbuch der Mathematik*. Teubner Verlag, Wiesbaden, 2003.
- [76] Harr, M. E.: *Foundations of Theoretical Soil Mechanics*. McGraw-Hill Book Company, New York, 1966.

- [77] Hilber, H.; Hughes, T.; Taylor, R.: Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering & Structural Dynamics*, 5, 283–292, 1977.
- [78] Hornbogen, Erhard; Eggeler, Gunther; Werner, Ewald: *Werkstoffe: Aufbau und Eigenschaften von Keramik-, Metall-, Polymer- und Verbundwerkstoffen*. Springer, Heidelberg, Dordrecht, London, New York, 2012.
- [79] Hughes, T. J. R.: *The Finite Element Method. Linear static and dynamic finite element analysis*. Dover Publications, Mineola, New York, 2000.
- [80] Institut für rechnergestützte Modellierung im Bauingenieurwesen, Technischen Universität Braunschweig: Aman High Performance Computing. <https://www.tu-braunschweig.de/hpc-fk3/aman>.
- [81] Johnson, Anthony N.: *Development of a Three Dimensional Perfectly Matched Layer for Transient Elasto-Dynamic Analysis*. Dissertation, Naval Postgraduate School, Monterey, California, 2006.
- [82] Jonsson, Isak; Kågström, Bo: Recursive blocked Algorithms for Solving Triangular Systems: Part I: One-Sided and Coupled Sylvester-type Matrix Equations. *ACM Transactions on Mathematical Software*, 28(4), 392–415, 2002.
- [83] Jonsson, Isak; Kågström, Bo: Recursive blocked Algorithms for Solving Triangular Systems: Part II: Two-Sided and Generalized Sylvester and Lyapunov Matrix Equations. *ACM Transactions on Mathematical Software*, 28(4), 416–435, 2002.
- [84] Jonsson, Isak; Kågström, Bo: RECSY — High Performance Library for Sylvester-Type Matrix Equations. online.
- [85] Karniadakis, George Em; Kirby II, Robert M.: *Parallel Scientific Computing in C++ and MPI: A Seamless Approach to Parallel Algorithms and Their Implementation*. Cambridge University Press, 2007.
- [86] Kirsch, F.; Klingmüller, O.: Erfahrungen aus 25 Jahren Pfahl-Integritätsprüfung in Deutschland – Ein Bericht aus dem Unterausschuss „Dynamische Pfahlprüfungen“ des Arbeitskreises 2.1 „Pfähle“ der Deutschen Gesellschaft für Geotechnik e.V. 80(9), 640–650, 2003.
- [87] Klein, Ralf: *Dynamische Interaktion von dünnwandigen Tragwerken und Boden mit Abschirmschlitzten*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 1996.
- [88] Koelbel, C. H.; Loveman, D. B.; Schreiber, R. S.; Steele Jr., G. L.; Zosel, M. E.: *The High Performance Fortran Handbook*. The MIT Press, 1997.
- [89] Kojić, M.; Bathe, K.-J.: *Inelastic Analysis of Solids and Structures*. Springer, Berlin, 2005.

- [90] Korenev, B.G.; Rabinovic, I.M.: *Baudynamik, Handbuch*. VEB Verlag Bauwesen, Berlin, 1980.
- [91] Kölner Verkehrs-Betriebe AG: Schallübertragung aus einem U-Bahn-Tunnel in den Dom. <http://www.kvb-koeln.de/german/nachrichten/view.html?action=shownews&page=1&id=1854>.
- [92] Lamport, Leslie: Time, Clocks, and the Ordering of Events in a Distributed System. *Communications of the ACM*, 21(7), 558–565, July 1978.
- [93] LAPACK: Linear Algebra PACKage. <http://www.netlib.org/lapack>.
- [94] Latz, Kersten: *Dynamische Interaktion von Flüssigkeitsbehältern und Baugrund*. Dissertation, Technische Universität Braunschweig, Braunschweiger Schriften zur Mechanik, 1994.
- [95] Lehmann, L.: Schnelles Verfahren zur Berechnung der Baugrund-Bauwerk-Interaktion im Zeitbereich. *D-A-CH Mitteilungsblatt*, 22, 6–9, 2003.
- [96] Lehmann, L.: An effective finite element approach for soil-structure analysis in the time-domain. *Structural Engineering and Mechanics*, 21, 437–450, 2005.
- [97] Lehmann, L.: Time Domain Analysis of Wave Propagation in Infinite Domains. *IAS-ME/WSEAS International Conference on Fluid Mechanics, Miami, USA*, 2006.
- [98] Lehmann, L.: *Wave Propagation in Infinite Domains*. Springer, Berlin / Heidelberg, 2006.
- [99] Lehmann, L.; Antes, H.; Schanz, M.: Transient analysis of soil-structure interaction problems: An effective FEM/SBFEM approach. In *Advanced Numerical Analyses of Solids and Structures, and Beyond*, Graz, Institute for Structural Analysis, Verlag der Technischen Universität Graz, 99–116, 2004.
- [100] Lehmann, L.; Borsutzky, R.: Fast three-dimensional seismic simulation in the time-domain with consideration of soil-structure interaction. *COMPDYN, Computational Methods in Structural Dynamics and Earthquake Engineering, Rethymno, Greece*, 2007.
- [101] Leipholz, H.: *Einführung in die Elastizitätstheorie*. G. Braun, Karlsruhe, 1968.
- [102] Li, Xiaoye S.: An Overview of SuperLU: Algorithms, Implementation, and User Interface. *ACM Transactions on Mathematical Software*, 31(3), 302–325, September 2005.
- [103] Li, Xiaoye S.; Demmel, James W.: SuperLU DIST: A Scalable Distributed-Memory Sparse Direct Solver for Unsymmetric Linear Systems. *ACM Trans. Mathematical Software*, 29(2), 110–140, June 2003.
- [104] Li, X.S.; Demmel, J.W.; Gilbert, J.R.; iL. Grigori; Shao, M.; Yamazaki, I.: SuperLU Users' Guide. Technical report LBNL-44289, Lawrence Berkeley National Laboratory, September 1999. <http://crd.lbl.gov/~xiaoye/SuperLU/>. Last update: August 2011.

- [105] Liao, Z. P.; Wong, H. L.: A transmitting boundary for the numerical simulation of elastic wave propagation. *Soil Dynamics and Earthquake Engineering*, 3(4), 174–183, 1984.
- [106] Lin, Chia-Ching; Lawton, E. C.; Caliendo, J. A.; Anderson, L. R.: An Iterative Finite Element-Boundary Element Algorithm. *Computers & Structures*, 39(5), 899–909, 1996.
- [107] Lin, Gao; Du, Jianguo; Hu, Zhiqiang: Earthquake analysis of arch and gravity dams including the effects of foundation inhomogeneity. *Frontiers of Architecture and Civil Engineering in China*, 1(1), 41–50, 2007.
- [108] Liu, J.; Lin, G.: Analysis of a Quadruple Corner-Cut Ridged/Vane-Loaded Circular Waveguide using Scaled Boundary Finite Element Method. *Progress In Electromagnetics Research*, 17, 113–133, 2011.
- [109] Liu, Jun; Lin, Gao; Wang, Fuming; Li, Jianbo: [The scaled boundary finite element method applied to electromagnetic field problems](#). *WCCM/APCOM IOP Conf. Series: Materials Science and Engineering*, (1), 2010.
- [110] Loveman, D. B.: [High performance Fortran](#). *Parallel & Distributed Technology: Systems & Applications*, *IEEE*, 1(1), 25–42, 1993.
- [111] Lurje, A. I.: *Räumliche Probleme der Elastizitätstheorie*. Akademie-Verlag GmbH, Berlin, 1963.
- [112] Lysmer, J.; Kuhlmeyer, R. L.: Finite dynamic model for infinite media. *Journal of Engineering Mechanics*, 95, 859–875, 1969.
- [113] Mahran, Essam: *Finite-Infinite-Elemente-Modellierung zur Simulation der Boden-Bauwerk-Flüssigkeit-Interaktion unter dynamischer Anregung*. Dissertation, Bergische Universität Wuppertal, Schriftenreihe des Instituts für Konstruktiven Ingenieurbau, Heft 3, 2004.
- [114] Manolis, G. D.; Beskos, D. E.: *Boundary Element Method in Elastodynamics*. Unwin Hyman Ltd, London, UK, 1988.
- [115] Meskouris, Konstantin; Hinzen, Klaus-G.; Butenweg, Christoph; Mistler, Michael: *Bauwerke und Erdbeben - Grundlagen - Anwendung - Beispiele*. Vieweg+Teubner Verlag, Wiesbaden, 2011.
- [116] Moore, Gordon E.: Cramming more components onto integrated circuits. *Electronics Magazine*, 38(8), 114–117, 1965.
- [117] MPI Forum: The Message Passing Interface (MPI) standard. <http://www.mcs.anl.gov/mpi>, 1994.
- [118] MUMPS Team: MUMPS - a MULTifrontal Massively Parallel sparse direct Solver. <http://graal.ens-lyon.fr/MUMPS>.

- [119] Müller, Kai: *Dreidimensionale dynamische Tunnel-Halbraum-Interaktion: Ein Verfahren auf der Grundlage einer Kopplung der Integraltransformationsmethode mit der Finite-Elemente-Methode*. Dissertation, Technische Universität München, Shaker-Verlag, Aachen, 2007.
- [120] Newmark, N.: A method of computation for structural dynamics. *Journal of Engineering Mechanics Division*, 85, 67–94, 1959.
- [121] OpenMP Forum: The OpenMP® API specification for parallel programming. <http://openmp.org>.
- [122] Petersen, Christan: *Dynamik der Baukonstruktionen*. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiesbaden, 2000.
- [123] Pläßmann, B.; Kirsch, F.; Löhr, M.; Vittinghoff, T.: Implementierung infiniten Elemente in ANSYS Open System und deren Anwendung bei Halbraumberechnungen in der Geotechnik. Technical report, 17. CAD-FEM Users Meeting, Sonthofen (Allgäu), 6.-8. Oktober 1999.
- [124] Pläßmann, Bernd: *Zur Optimierung der Meßtechnik und der Auswertemethodik bei Pfahlintegritätsprüfungen*. Dissertation, Technische Universität Braunschweig, 2002.
- [125] Poulos, H. G.; Davis, E. H.: *Elastic Solutions for Soil and Rock Mechanics*. John Wiley & Sons, INC., New York, London, Sydney, Toronto, 1974.
- [126] Prabucki, M. J.; von Estorff, O.: [Dynamic response in the time domain by coupled boundary and finite elements](#). *Computational Mechanics*, (6), 35–46, 1990.
- [127] Protopapadakis, E. E.; Schauer, M.; Pierri, E.; Doulamis, A. D.; Stavroulakis, G. E.; Böhrnsen, J.-U.; Langer, S. C.: Evaluation of pile integrity tests using coupled FEM/SBFEM simulation and genetically optimized neural classifiers. 2014 unveröffentlicht.
- [128] Radmanović, B.; Katz, C.: [A High Performance Scaled Boundary Finite Element Method](#). *IOP Conf. Series: Material Science and Engineering*, 10, 2010.
- [129] Reinhardt, H. J.: *Numerik gewöhnlicher Differentialgleichungen: Anfangs- und Randwertprobleme*. De Gruyter Lehrbuch Series, Walter de Gruyter, 2008.
- [130] Renault, Philippe Lucien Arno: *Bewertungsverfahren zur Beurteilung der Erdbbensicherheit von Brückenbauwerken*. Dissertation, Reinisch - Westfälische Technische Hochschule Aachen, baustatik - Baudynamik, 2007.
- [131] Richart, F.E.; Woods, R. D.; Hall, J. R.: *Vibrations of soils and foundations*. Prentice-Hall International Series in Theoretical and Applied Mechanics, Englewood Cliffs/N.J., 1970.

- [132] Roduner, Ch.: Die Riccati-Gleichung. *ETH Zürich, Institut für Mess- und Regeltechnik IMTR-Bericht*, 26, 1994.
- [133] Rostásy, Ferdinand S.: *Baustoffe*. Verlag W. Kohlhammer, Stuttgart; Berlin; Köln; Mainz, 1983.
- [134] Rurkowska, Katherina; Schauer, Marco; Langer, Sabine: [On Neumann - Dirchlet coupling strategy of Finite Element and Boundary Element method in elastodynamics](#). *Proceedings in Applied Mathematics and Mechanics*, 11, 265–266, 2011.
- [135] Saad, Yousef: *Iterative Methods for Sparse Linear Systems*. Copyright by Yousef Saad, second edition with corrections, 2000.
- [136] ScaLAPACK: Scalable Linear Algebra PACKage. <http://www.netlib.org/scalapack>.
- [137] Schaller, Robert R.: [Moore's Law: past, present and future](#). *IEEE Spectrum*, 34(6), 52–59, 1997.
- [138] Schauer, M.; Langer, S.: [Numerical Simulations of Pile Integrity Tests Using a Coupled FEM/SBFEM Approach](#). *Proceedings in Applied Mathematics and Mechanics*, 12, 547–548, 2012.
- [139] Schauer, M.; Lehmann, L.: [Large Scale Simulation with Scaled Boundary Finite Element Method](#). *Proceedings in Applied Mathematics and Mechanics*, 9(1), 103–106, 2009.
- [140] Schauer, M.; Psychas, I. D.; Böhrnsen, J.-U.; Marinaki, M.; Marinakis, Y.; Langer, S. C.; Stavroulakis, G. E.: Detection and Differentiation of Defective Pile Geometries using a Coupled FEM/SBFEM Approach and an Ant Colony Classification Algorithm. 2014 unveröffentlicht.
- [141] Schauer, Marco; Langer, Sabine; Roman, Jose E.; Quintana-Ortí, Enrique S.: [Large scale simulation of wave propagation in soils interacting with structures using FEM and SBFEM](#). *Journal of Computational Acoustics*, 19(1), 75–93, 2011.
- [142] Schauer, Marco; Roman, Jose E.: SCABO Manual. Technical report Revision: 369, TU Braunschweig, Institut für Angewandte Mechanik (2010–2013), Institut für Konstruktionstechnik (seit 2013), 2010–2014. <https://www.tu-braunschweig.de/ik>.
- [143] Schauer, Marco; Roman, Jose E.; Quintana-Ortí, Enrique S.; Langer, Sabine: [Parallel Computation of 3-D Soil-Structure Interaction in Time Domain with a Coupled FEM/SBFEM Approach](#). *Journal of Scientific Computing*, 52, 446–467, 2012.
- [144] Schot, Steven H.: Eighty Years of Sommerfeld's Radiation Condition. *Historia Mathematica*, 19, 385–401, 1992.
- [145] Shah, Siddharth G.; Solanki, C. H.; Desai, J. A.: Soil structure interaction analysis methods - State of art-Review. *International Journal of Civil and Structural Engineering*, 2(1), 176–204, 2011.

- [146] SLICOT: The Control and Systems Library SLICOT. www.slicot.org.
- [147] Soares Jr., D.; Mansur, W. J.; von Estorff, O.: [An efficient time-domain FEM/BEM coupling approach based on FEM implicit Green's functions and truncation of BEM time convolution process](#). *Comput. Methods Appl. Mech. Engrg.*, 196, 1816–1826, 2007.
- [148] Sommerfeld, Arnold: *Lectures on theoretical physics Vol. 6: Partial differential equations in physics*. Academic Press, New-York, 1949.
- [149] Song, Chongmin: [The scaled boundary finite-element method in structural dynamics](#). *International Journal for Numerical Methods in Engineering*, 77, 1139–1171, 2009.
- [150] Song, Chongmin; Wolf, John P.: [Semi-analytical representation of stress singularities as occurring in cracks in anisotropic multi-materials with the scaled boundary finite-element method](#). *Computers and Structures*, 80(2), 183–197, 2002.
- [151] Storti, M.; Nigro, N.; Paz, R.; Dalcín, L.; Battaglia, L.; López, E.; Ríos Rodríguez, G.A.: PETSc-FEM, A General Purpose, Parallel, Multi-Physics FEM Program. Technical report, 1999-2010. <http://www.cimec.org.ar/petscfem>.
- [152] Strutt, J. W. (Lord Rayleigh): *The Theory of Sound*, Vol. 1. MacMillan and Co, London, 1877.
- [153] Strutt, J. W. (Lord Rayleigh): On waves propagated along the plane surface of an elastic solid. *Proceedings of the London Mathematical Society*, 17(1), 4–11, 1885.
- [154] Sun, Xian-He; Chen, Yong: Reevaluating Amdahl's law in the multicore era. *Journal of Parallel and Distributed Computing*, 70, 183–188, 2010.
- [155] Sun, Xian-He; Ni, Lionel M.: Scalable Problems and Memory-Bounded Speedup. *Journal of Parallel and Distributed Computing*, 19, 27–37, 1993.
- [156] Trinks, Carolin: *Consistent absorbing boundaries for time-domain interaction analyses using the fractional calculus*. Dissertation, Technische Universität Dresden, Veröffentlichungen des Lehrstuhls für Statik, 2005.
- [157] Vaish, A. K.; Chopra, A. K.: Earthquake Finite Element Analysis of Structure-Foundation Systems. *Journal of the Engineering Mechanics Division*, 100, 1101–1116, 1974.
- [158] Voigt, W.: *Lehrbuch der Kristallphysik*. B.G. Teubners Sammlung von Lehrbüchern auf dem Gebiete der mathematischen Wissenschaften mit Einschluss ihrer Anwendungen. B.G. Teubner, 1910.
- [159] von Estorff, O.; Firuziaan, M.: Coupled BEM/FEM approach for nonlinear soil/structure interaction. *Engineering Analysis with Boundary Elements*, 24, 715–725, 2000.
- [160] Wilson, John Marius: Imperial Gazetteer of England; (1870-72), Wales: Manchester. <http://www.genuki.org.uk/big/eng/LAN/Manchester/ImpGaz1872.shtml>.

- [161] Wolf, J.: *The Scaled Boundary Finite Element Method*. John Wiley & Sons, Chichester, 2003.
- [162] Wolf, J.; Song, C.: *Finite-Element Modelling of Unbounded Media*. John Wiley & Sons, Chichester, 1996.
- [163] Wolf, John P.: [Response of unbounded soil in scaled boundary finite-element method](#). *Earthquake Engineering & Structural Dynamics*, 31(1), 15–32, 2002.
- [164] Wong, H. L.; Luco, J. E.: [Dynamic response of rigid foundations of arbitrary shape](#). *Earthquake Engineering & Structural Dynamics*, 4(6), 579–587, 1976.
- [165] Wood, W. L.; Bossak, M.; Zienkiewicz, O. C.: An Alpha Modification of Newmark's Method. *Internat. J. Numer. Methods Engrg.*, 15, 1562–1566, 1981.
- [166] Wulkau, M.; Langer, S.: [Virtual test facilities for building acoustic predictions using FEM](#). *Proceedings in Applied Mathematics and Mechanics*, 11, 643–644, 2011.
- [167] Yang, Z.J.; Deeks, A.J.; Hao, H.: [Transient dynamic fracture analysis using scaled boundary finite element method: a frequency-domain approach](#). *Engineering Fracture Mechanics*, 74, 669–687, 2007.
- [168] Zhang, X.; Wegner, J. L.; Haddow, J. B.: Three-Dimensional Dynamic Soil-Structure Interaction Analysis in the Time Domain. *Earthquake Engineering and Structural Dynamics*, 28(12), 1501–1524, Dezember 1999.
- [169] Zienkiewicz, O. C.; Kelly, D. M.; Bettess, P.: The Coupling of the Finite Element Method and Boundary Solution Procedures. *International Journal for Numerical Methods*, 11, 355–375, 1977.
- [170] Zienkiewicz, O. C.; Taylor, R. L.: *The Finite Elemente Method for Solid and Structural Mechanics*, 6th edition. Elsevier Butterworth-Heinemann, Oxford, 2005.
- [171] Zienkiewicz, O. C.; Taylor, R. L.; Zhu, J. Z.: *The Finite Elemente Method: Its Basis & Fundamentals*, 6th edition. Elsevier Butterworth-Heinemann, Oxford, 2005.

Berichte aus dem Institut für Statik ab 2003

- Nr. 2003-96 B. HÜBNER: Simultane Analyse von Bauwerk-Wind-Wechselwirkungen
- Nr. 2003-97 J. GEISTEFELDT: Stochastische Finite-Element-Methoden mit Anwendung auf aeroelastische Tragsysteme
- Nr. 2003-98 O. KNOKE: Beulwiderstände zusammengesetzter Zylinder-Kegel-Schalen
- Nr. 2005-99 A. KÖLKE: Modellierung und Diskretisierung bewegter Diskontinuitäten in randgekoppelten Mehrfeldsystemen
- Nr. 2006-100 D. DINKLER (HRSG.): Institut für Statik, Lehre und Forschung, 1996-2006
- Nr. 2006-101 A. VEHRE: Ein Reduktionsverfahren für Fluid-Struktur-Wechselwirkung mit Finiten Raum-Zeit-Elementen
- Nr. 2006-102 M. LÖHR: Analyse aeroelastischer Systeme mit Spektralen Stochastischen Finite-Element-Methoden
- Nr. 2006-103 L. ASCHENBRENNER: Mehrkomponenten-Modell zur Beschreibung des Deformationsverhaltens von Asphalt
- Nr. 2006-104 T. ZÜMENDORF: Ein gradientenabhängiges Modell für Schädigung bei viskoplastischem Materialverhalten
- Nr. 2007-105 C. LEPPERT: Mehrphasenmodell für granulare Medien zur numerischen Untersuchung des Phasenübergangs bei der Entleerung von Silos
- Nr. 2007-106 V. KRASE: Stability of Municipal Solid Waste Landfills
- Nr. 2007-107 J. PONTOW: Imperfektionsempfindlichkeit und Grenzlaster von Schalentragswerken
- Nr. 2009-108 M. BECKMANN: Ein gradientenabhängiges Modell für anisotrope Schädigung von Beton unter Berücksichtigung von Porendruck
- Nr. 2010-109 P. SUN: Fluid-Struktur-Wechselwirkung mit aktiver Schwingungskontrolle durch piezoelektrische Materialien
- Nr. 2010-110 J. VELDE: 3D Nonlocal Damage Modeling for Steel Structures under Earthquake Loading
- Nr. 2010-111 K. SCHUSTER: Systemidentifikation und Bauwerksüberwachung mit piezokeramischen Aktuatoren
- Nr. 2011-112 S. BENTE: Interaction of Degradation, Deformation and Transport Processes in Municipal Solid Waste Landfills
- Nr. 2011-113 L. OSTERMANN: Hochtemperaturverhalten von Beton – Gekoppelte Mehrfeld-Modellierung und numerische Analyse
- Nr. 2012-114 B. BRODERSEN: Modellierung des thermo-mechanischen Verbundverhaltens von Asphalt in Straßenaufbauten
- Nr. 2012-115 J. KINDLEIN: Gekoppelte Analyse von Reaktions- und Transportprozessen in Deponiestrukturen
- Nr. 2015-116 F. PASENOW: Modellierung oberflächengekoppelter Mehrfeldsysteme und numerische Analyse rutschender Bodenmaterialien

Sonderdrucke

Phänomenologische Modelle für Werkstoffe des Bauwesens.

Hermann Ahrens zum 60. Geburtstag (1998).

Herausgeber: D. Dinkler, U. Kowalsky.

Baustatik-Baupraxis 8.

Berichte der Fachtagung am 21. und 22. März 2002 in Braunschweig.

Herausgeber: D. Dinkler.

Institut für Statik, Technische Universität Braunschweig

Beethovenstraße 51, 38106 Braunschweig, Deutschland

Telefon +49 (0)531/391-3667, Telefax +49 (0)531/391-8116

E-Mail statik@tu-bs.de, Homepage <https://www.tu-braunschweig.de/statik>